This document is based on European standards and is not valid for use in U.S.A.

Compact / CANopen / Logic Controller / M238 + Optimized Hoisting

System User Guide

EIO0000000295

MAY 2010

Schneider Electric

Contents

Important Information

NOTICE Read these instructions carefully, and study the equipment to become familiar with the device before trying to install, operate, or maintain it. The following types of special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, that will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation that, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation that, if not avoided, **can result in** death, serious injury, or equipment damage.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation that, if not avoided, **can result in** injury or equipment damage.

PLEASE NOTE Electrical equipment should only be installed, operated, serviced, and maintained by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is a person who has skills and knowledge related to the construction and operation of electrical equipment and the installation, and has received safety training to recognize and avoid the hazards involved

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to **the operator of that machine.**

⚠ WARNING

UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY

- Do not use this software and related automation products on equipment which does not have point-of-operation protection.
- Do not reach into machine during operation.

Failure to follow these instructions can cause death, serious injury or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation and maintenance of the machine; therefore, only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to the applicable local and national standards and regulations. A "National Safety Council's" Accident Prevention Manual also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products by itself cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks for point-of-operation protection have been installed and are operational before placing the equipment into service. All mechanical/electrical interlocks and safeties for point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Co-ordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of this document.

START UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters and debris from equipment.

Failure to follow these instructions can result in injury or equipment damage.

Follow all start up tests recommended in the equipment documentation. Store all equipment documentation for future reference.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove ground from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and rating of components, there are hazards that can be encountered if such equipment is improperly operated.

- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.

- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

⚠ WARNING

UNEXPECTED EQUIPMENT OPERATION

- Only use software tools approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can cause death, serious injury or equipment damage.

Introduction

Introduction

This document is intended to provide a quick introduction to the described system. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers additional information to the product documentation, for installing, configuring and implementing the system.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider Electric and third-party components may be integrated to fulfill an industrial application.

A detailed functional description or the specification for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The architecture described in this document has been fully tested in our laboratories using all the specific references you will find in the component list near the end of this document. Of course, your specific application requirements may be different and will require additional and/or different components. In this case, you will have to adapt the information provided in this document to your particular needs. To do so, you will need to consult the specific product documentation of the components that you are substituting in this architecture. Pay particular attention in conforming to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

It should be noted that there are some major components in the architecture described in this document that cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified herein. You must be aware of the consequences of component substitution in the architecture described in this document as substitutions may impair the compatibility and interoperability of software and hardware.

⚠ CAUTION

EQUIPMENT INCOMPATIBILITY OR INOPERABLE EQUIPMENT

Read and thoroughly understand all hardware and software documentation before attempting any component substitutions.

Failure to follow these instructions can result in injury or equipment damage.

This document describes a hoisting architecture based on Modicon M238 Logic controller S-Type.

Abbreviations

Abbreviation	Signification
AC	Alternating Current
CB	Circuit Breaker
CFC	Continuous Function Chart – a programming language based on function chart
DI	Digital Input
DO	Digital Output
DC	Direct Current
DFB	Derived Function Blocks
EDS	Electronic Data Sheet
E-STOP	Emergency Stop
FBD	Function Block Diagram – an IEC-61131 programming language
HMI	Human Machine Interface
I/O	Input/Output
IL	Instruction List - a textual IEC-61131 programming language
IP	Internet Protocol
LD	Ladder Diagram – a graphic IEC-61131 programming language
MBTCP	Communications protocol with Modbus over TCP (Ethernet)
MFB	PLCopen Motion Function Block
PC	Personal Computer
POU	Programmable Object Unit, Program Section in SoMachine
PDO	Process Data Object (CANopen)
PS	Power Supply
RMS	Root Mean Square
RPM	Revolution Per Minute
RTU	Remote Terminal Unit
RPDO	Receive Process Data Object (CANopen)
SD	Stepper Drive
SE	Schneider Electric
SFC	Sequential Function Chart – an IEC-61131 programming language
SDO	Service Data Object
ST	Structured Text – an IEC-61131 programming language
TCP	Transmission Control Protocol
TPDO	Transmit Process Data Object (CANopen)
TVDA	Tested, Validated and Documented Architecture
UDP	User Data Protocol
VSD	Variable Speed Drive
WxHxD	Dimensions : Width, Height and Depth

Glossary

Expression	Signification
Altivar (ATV)	SE product name for a family of VSDs
CANopen	Name for a communications machine bus system
ConneXium	SE product name for a Family of Transparent Factory devices
Harmony	SE product name for a family of switches and indicators
Magelis	SE product name for a family of HMI-Devices
OsiSense	SE product name for a family of sensors
Phaseo	SE product name for a family of power supplies
PLCopen	An international standard for industrial controller programming.
Preventa	SE product name for a family of safety devices
SoMachine	SE product name for an integrated software tool
TeSys	SE product name for a family of motor protection devices and load contactors
Vijeo Designer	An SE software product for programming Magelis HMI devices

Application Source Code

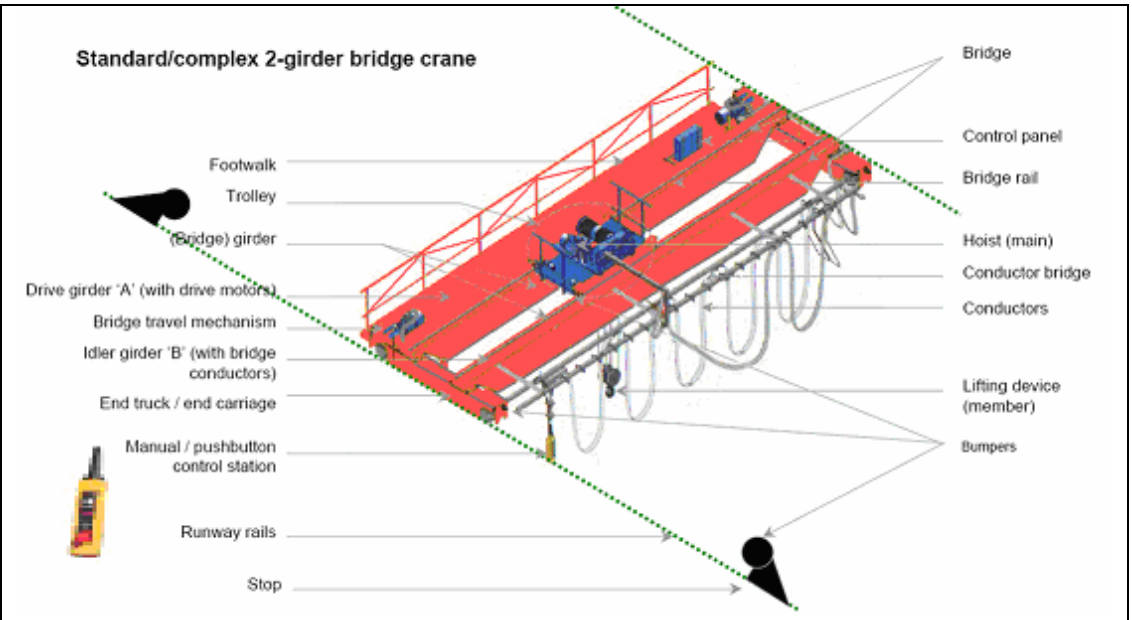
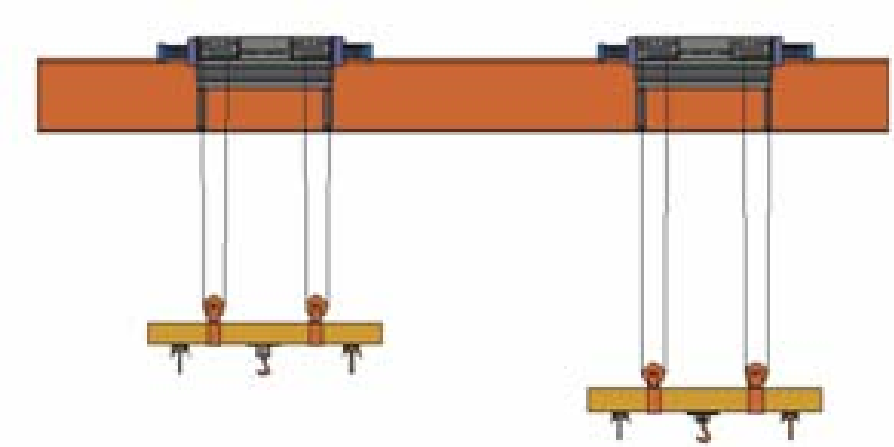
Introduction

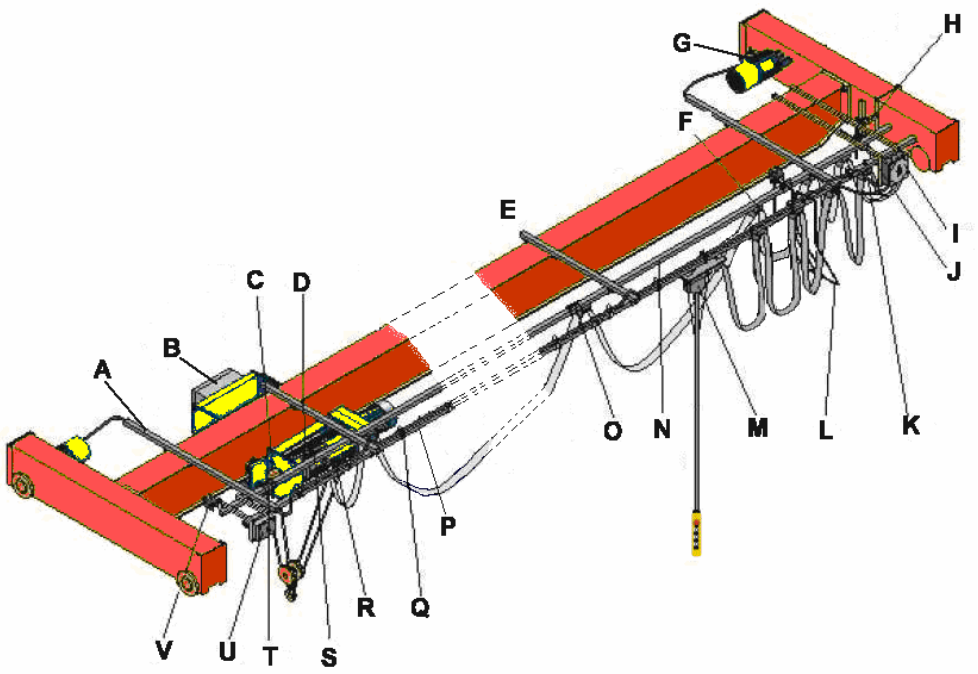
Examples of the source code and wiring diagrams used to attain the system function as described in this document can be downloaded from our website (registration is required, contact your Schneider Electric Application Design Expert).

The example source code is in the form of configuration, application and import files. Use the appropriate software tool to either open or import the files.

Extension	File Type	Software Tool Required
CSV	Comma Separated Values, Spreadsheet	MS Excel
DOC	Document file	Microsoft Word
DOP	Project File	Vijeo Designer Lite
DWG	Project file	AutoCAD
EDS	Electronic Data Sheet – Device Definition	Industrial standard
PDF	Portable Document Format - document	Adobe Acrobat
PROJECT	Project file	SoMachine
VDZ	Project file	Vijeo Designer
Z13	Project archive file	EPLAN

Example of the Typical Machine

Gantry Crane	
Industry Two Trolley crane	

Standard girder bridge crane	
	A) Profile support B) Hoist cabinet C) Limit switch for the end of the lifting height restriction bar and safety limit bar D) Carriage power feed traveling arm E) Profile fixing support to the girder F) Cable carriers G) Motor connection to the control cabinet via fast plug connectors H) Power point arm for crane power feed and connection boxes I) Bridge travel cabinet J) Bridge travel limit switch K) End limit L) Festoon cable protection supports M) Pendant plug-in connector N) Supports for trolley and pendant power feed systems O) Profile connection P) Sliding support Q) Cable clips fitted to the profile for supporting extra electric cables R) Load Limit device S) Trolley limit switch T) Limit of carriage cross travel running U) Radio (optional) V) Fixed beam trolley limit

Hoisting Solution

Introduction

This chapter describes the function, the architecture, dimensions, quantities and the different types of components used within this solution.

General

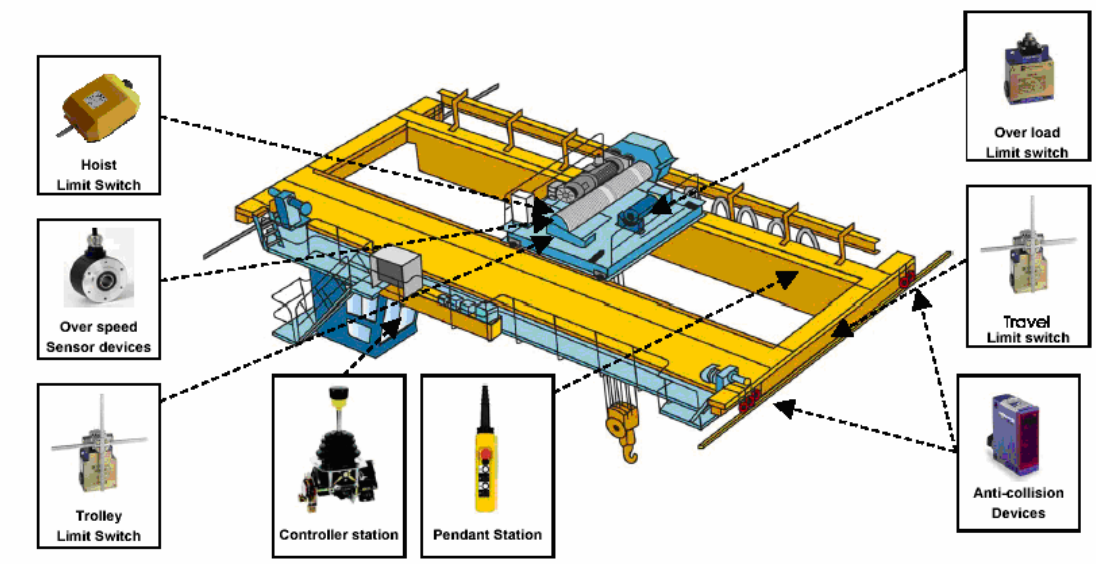
The controller in this application is a Modicon M238 Logic controller. The drives ATV71 and ATV312 are connected to the controller via a CANopen bus. The example application includes two functional safety options according to EN ISO 13849-1 standards: an Emergency Stop function supervised by a Preventa Safety Module (see the appropriate hardware manual), plus a second Preventa Safety Module to evaluate protective door sensors.

Operational Function

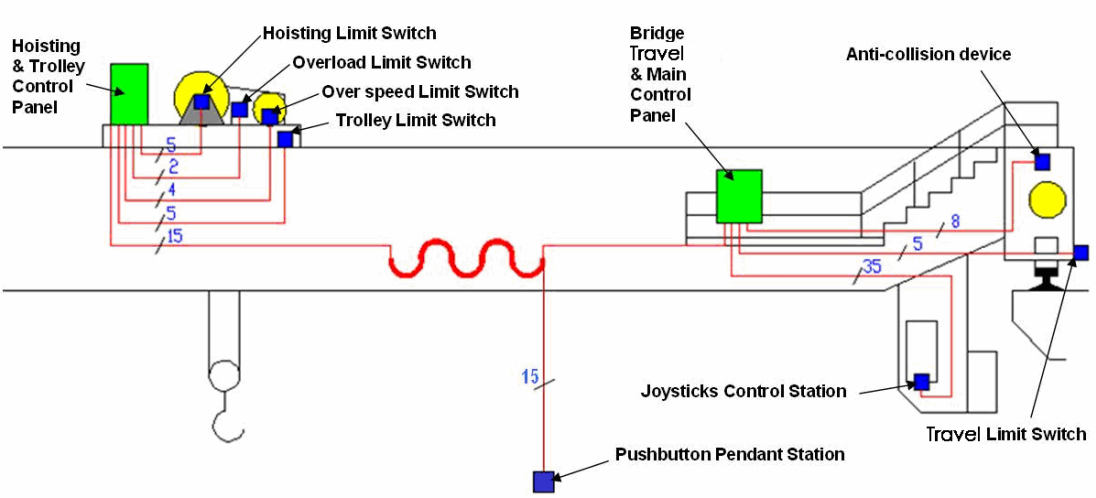
Introduction

- The **Industry crane** has a dual 3 axis movement. 2 x Hoist, 2 x Trolley and 2 x Traveling. The movement is performed using variable speed drives.
- The operator controls the crane using a pendant station or, optionally, a radio remote control station. The control device has the power on/off switches.
- For bridge travel and trolley movements, a minimum of two speeds are implemented.
- To help protect the crane from damage, all movement is monitored with sensors for indication or detection. The controller software uses the input from these devices as inputs to the **A**pplication **F**unction **B**locks which are described in further detail within this document.

Explanation of Crane movements



Automation Device Placement



Application Functions

Schneider Electric provides an **A**pplication **F**unction **B**lock Library (AFB) for Hoisting. This is a set of function blocks, tested & validated for functionality and has been designed to help OEMs develop their crane applications.

Schneider Electric offers a set of function blocks (FBs) that provide various application and device functions. They do not provide a complete user program for a crane, but they can save the OEM a lot of development effort.

The following is a list of AFB functions for the gantry crane of type Hoisting Industry:

Hoist Movement:

- Limit switch management
- Overload control
- Load overspeed control
- Speed optimization and rope slack
- Hoisting position synchronization

Trolley Movement:

- Anti-sway
- Limit switch management

Traveling Movement:

- Limit switch management
- Anti-crab
- Anti-sway

General:

- Speed select
- Scale input
- Monitoring Data Storage

For additional information concerning the Hoisting AFBs please refer to the SoMachine online help.

Architecture

General

The controller is a Modicon M238 Logic controller (S-Type).
The user can control the application using the pendant control station.

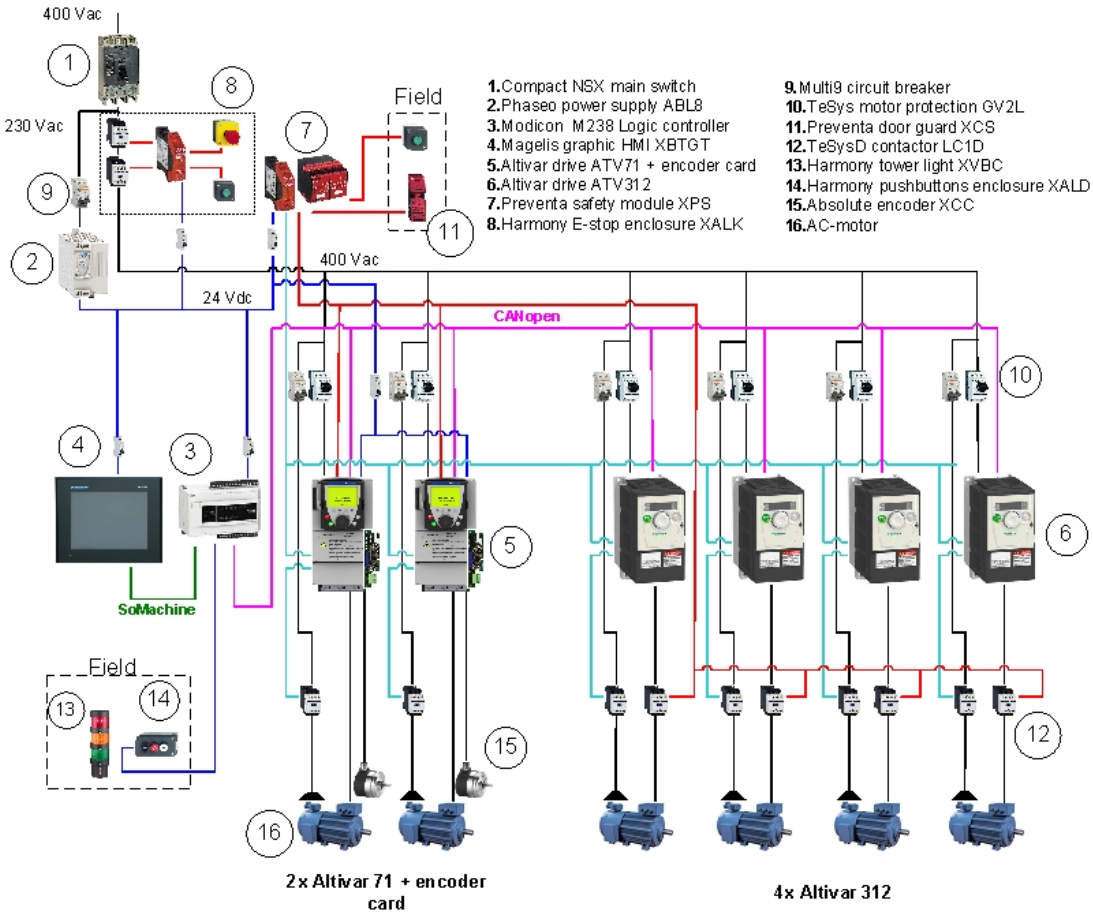
The Altivar 71 and Altivar 312 drives are connected via CANopen fieldbus to the M238.

Field devices such as encoder, limit switches and proximity sensors monitor position.

The HMI shows only the crane status.

The example application includes an option for an Emergency Stop function supervised by a Preventa safety module.

Layout



Components

Hardware:

- Compact NSX100F main switch
- Phaseo ABL8 power supply
- Modicon M238 Logic controller (S- type)
- Magelis XBTGT2330 HMI
- Preventa safety relay XPSAF
- Altivar 71 variable speed drive
- Altivar 312 variable speed drive
- TeSysD load contactors LC1D
- TeSys GV2 motor circuit breaker
- OsiSense Sensors
- Harmony XACA Pendant Station
- Harmony tower light XVBC
- Limit Switch XCKMR
- Screw Limit Switch (TER-International)
- OsiSense XCC Encoder
- Force sensor (load cell from Vishay)

Software:

- SoMachine V2.0

Quantities of Components

For a complete and detailed list of components, the quantities required and the order numbers please refer to the components list at the rear of this document.

Degree of Protection

Not all the components in this configuration are designed to withstand the same environmental conditions. Some components may need additional protection, such as housings, depending on the environment in which you intend to use them. For environmental details of the individual components please refer to the list in the appendix of this document and the corresponding user manual.

Cabinet Technical Data

Input	Mains voltage	400 Vac
	Power requirement	~ 4.5 kW
	Cable Size	5 x 2.5 mm² (L1, L2, L3, N, PE)
	Cable Connection	3 phase + Neutral + Ground Neutral is needed for 230 Vac (Phase and Neutral)
Output	Motor power ratings	4 asynchronous motors (4 poles:1500 RPM) controlled by ATV312 (0.75 kW) 2 asynchronous motors (4 poles:1500 RPM) controlled by ATV71 (0.75 kW)

Functional Safety Notice

(EN ISO 13849-1 EN IEC 62061)

The standard and level of functional safety you apply to your application is determined by your system design and the overall extent to which your system may be a hazard to people and machinery.

As there are no moving mechanical parts in this application example, category 1 (according to EN ISO 13849-1) has been selected as an optional safety level.

Whether or not this functional safety category should be applied to your system should be ascertained with a proper risk analysis.

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the functional safety requirements with respect to the equipment used in their systems or of compliance with either national or international safety laws or regulations.

Emergency Stop

Emergency Stop/Emergency Disconnection function

This function for stopping in an emergency is a protective measure which compliments the safety functions for the safeguarding of hazardous zones according to prEN ISO 12100-2.

Safety Functions

Door guarding

up to Performance Level (PL) = b, Category 1, Safety Integrity Level (SIL) = 1

Dimensions

The dimensions of the individual devices used; controller, drive, power supply, etc. require a main cabinet size of at least 1000 x 1800 x 600 mm (WxHxD).

The HMI display, illuminated indicators such as ”SYSTEM ON“, ”SYSTEM OFF“ or ”ACKNOWLEDGE EMERGENCY STOP“ as well as the Emergency Stop switch itself, can be built into the door of the cabinet.

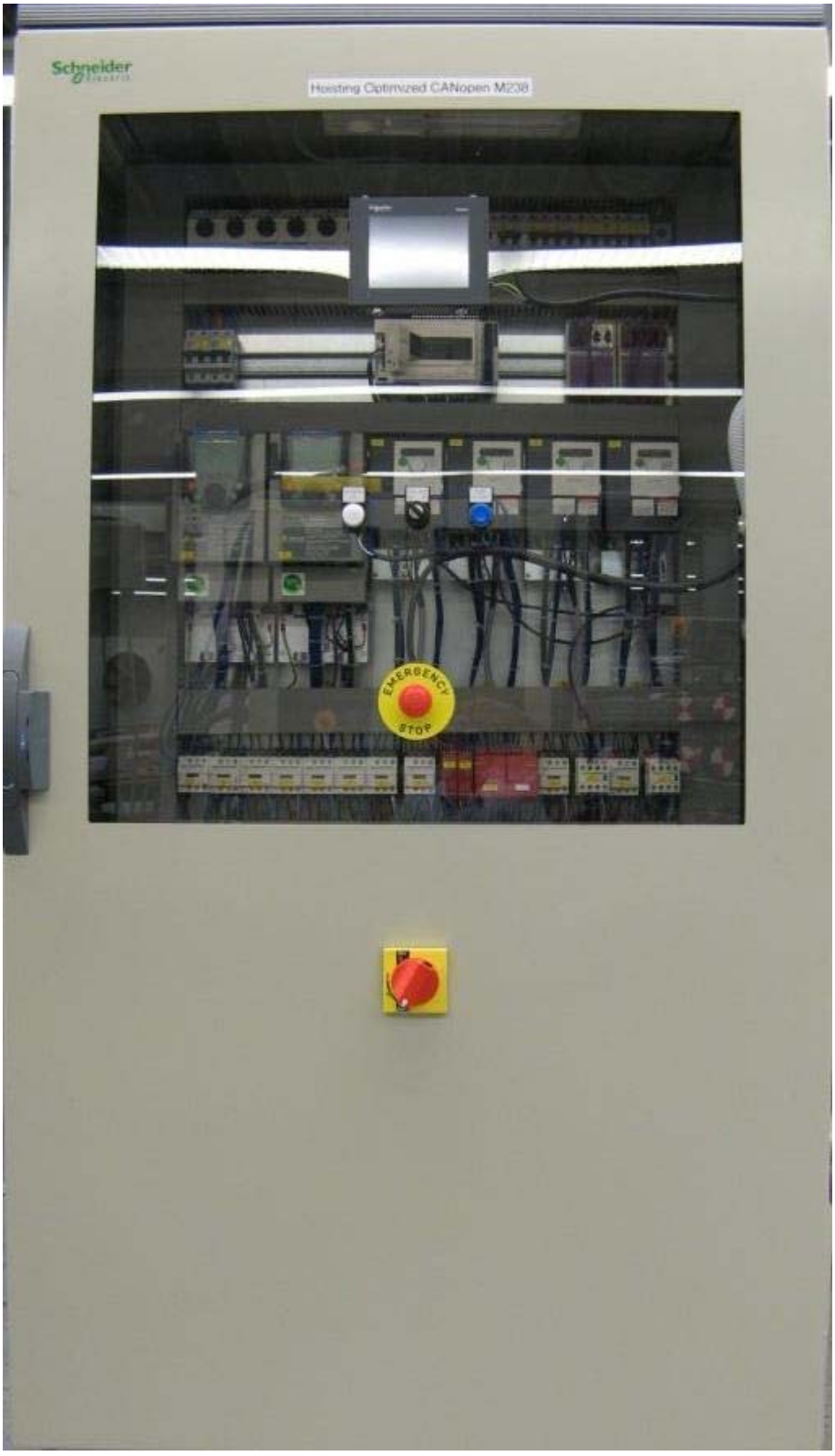
Installation

Introduction

This chapter describes the steps necessary to set up and assemble the hardware and configure the software required to implement the application.

Assembly

Main cabinet front



Main cabinet interior

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

NOTE

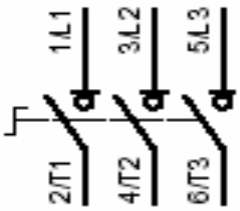
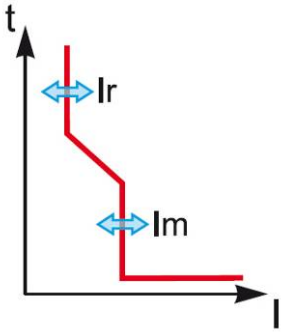
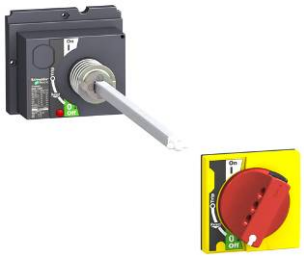
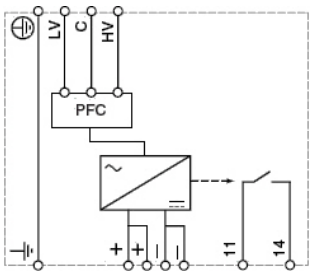
A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and the installation, and has received safety training to recognize and avoid the hazards involved

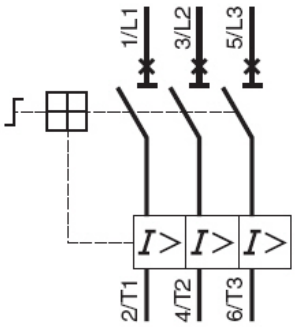
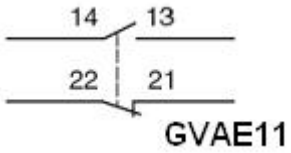
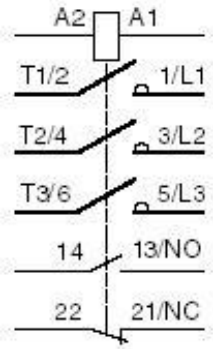
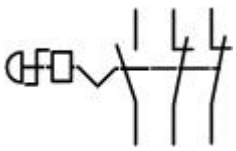
The architecture will have to be redesigned to match power requirements.
This includes the drives, the cabling, the switches and the contactors.

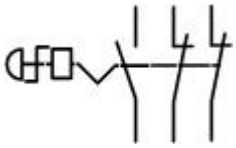
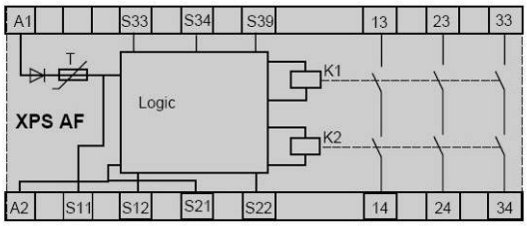
Hardware

General

General description of the hardware.

Main Switch **Compact NSX100F** **LV429003** 36 kA 380/415 Vac		
Main Switch **Compact NSX100F** **LV429035** **Trip unit TM32D** **Thermal-magnetic** 32 A		 Ir - Thermal protection Im - Magnetic protection
Main Switch **Compact NSX100F** **Rotary handle** **LV429340** **Terminal shield** **LV429515**	 Rotary handle with red handle on yellow front	 Terminal shield short
Power supply **Phaseo** **ABL8RPS24100** Primary 200…500 Vac, Secondary 24 Vdc, 240 W, 10 A		

Motor Circuit Breaker **GV2L08** **and** **GV2L10** **with** **auxiliary contact** **GVAE11**	 	 
Load Contactor **TeSysD** **LC1D18BL**		
Circuit Breaker **Multi 9** **23726** **23747** **23756** **24427** **24444**	 	 
Emergency Stop **Switch** **(trigger action)** **Harmony** **XALK178G**		

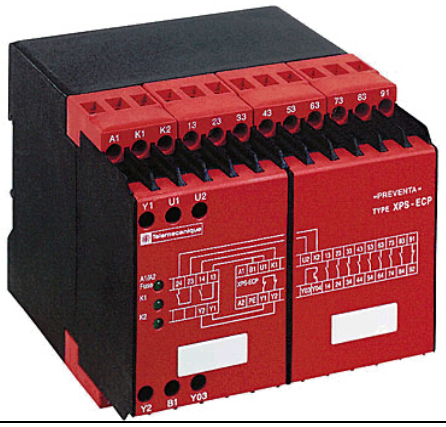
Emergency Stop **Harmony** **XB5AS844 + B5AZ141** **Including Label ZBY8330**		
Safety Module **Preventa** **XPSAF5130**	 	

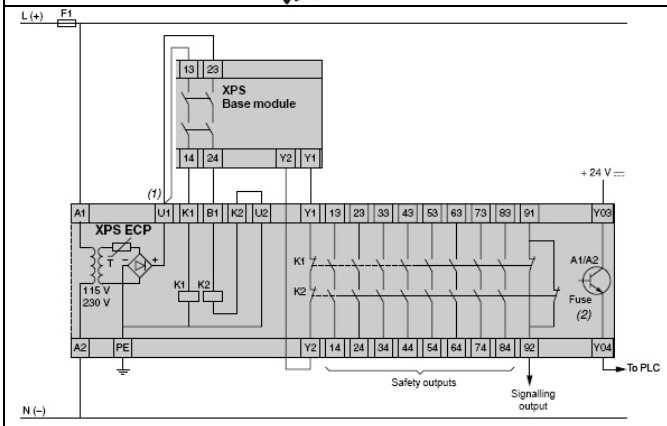
Expansion Module

Preventa

XPSECP5131

to increase the number of safety output contacts of the base module.





(1) When installing base modules and modules for increasing the number of safety contacts into different electrical enclosures, run separate cables for terminals U1-13 and U1-23.

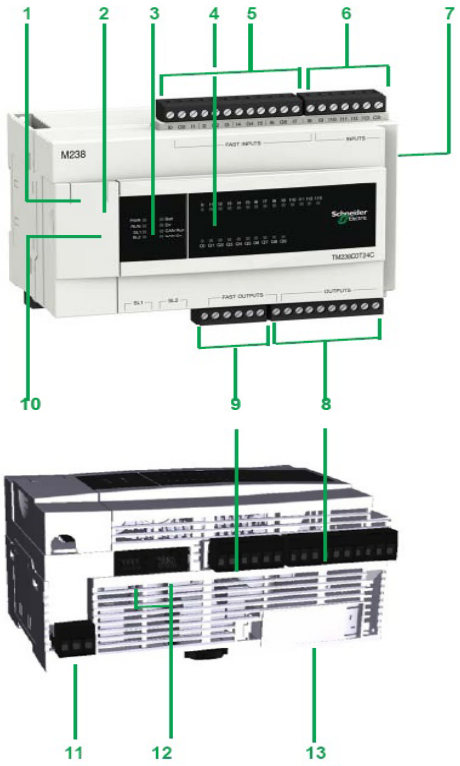
(2) Operating status of internal electronic fuse.

Modicon M238

Logic Controller

TM238LFDC24DTS0

14 Digital Input
10 Digital Output



1. Mini B USB port, for a programming terminal.

2. A hinged access cover with 2 cable glands (1 removable for the terminal cord set and 1 for the CANopen cable).

3. The controller status by means of 4 LEDs (PWR, RUN, Batt and Err) The integrated communication port status by means of 4 LEDs (SL1, SL2,CAN Run and CAN Err.).

4. A display unit showing the I/O states (I0..I13 and Q0..Q9).

5. A removable screw terminal block (12 terminals) for connecting the sensors (24 Vdc fast inputs).

6. A removable screw terminal block (7 terminals) for connecting the sensors (24 Vdc inputs).

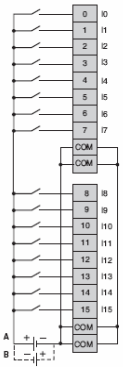
7. A connector for discrete TM2D●●, analog TM2A●● and counter TM200HSC210D● I/O extension modules (7 modules max.).

8. A removable screw terminal block (10 terminals) for connecting 6 pre-actuators (24 Vdc outputs).

Modicon Logic controller M238

TM2DDI16DT

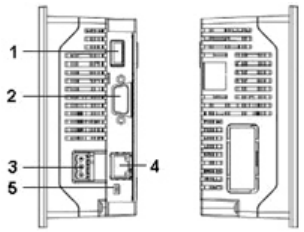
16 Digital Inputs



Magelis HMI

XBTGT2330

24 Vdc Input, TFT Color LCD, 320 x 240 Pixels, 65536 Colors, 16 MB Application Flash EPROM with Built-in Ethernet



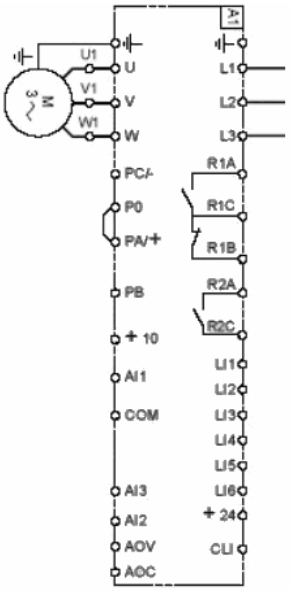
1. **USB- interface**
2. **Serial interface COM1**
3. **Power supply connector**
4. **Serial interface COM2**
5. **(RJ45) Polarization Switch**

Altivar 312

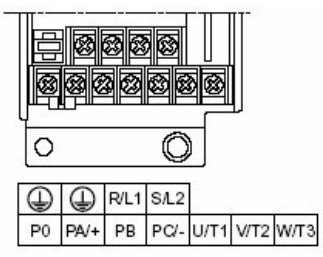
ATV312H075N4

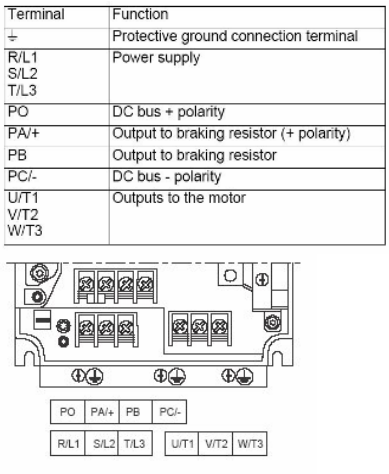
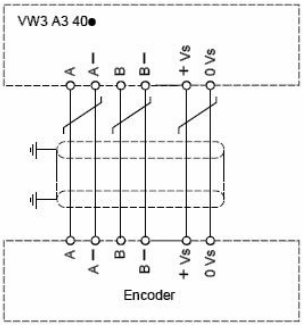
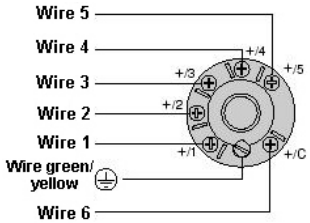
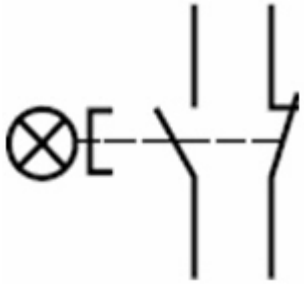
3-phase

400 Vac, 0.75 kW



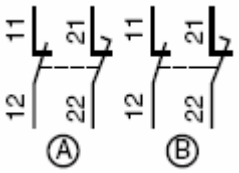
Terminal	Function
⏚	Ground terminal
R/L1 S/L2	Power supply
R/L1 S/L2 T/L3	
PO	DC bus + polarity
PA/+	Output to braking resistor (+ polarity)
PB	Output to braking resistor
PC/-	DC bus - polarity
U/T1 V/T2 W/T3	Outputs to the motor



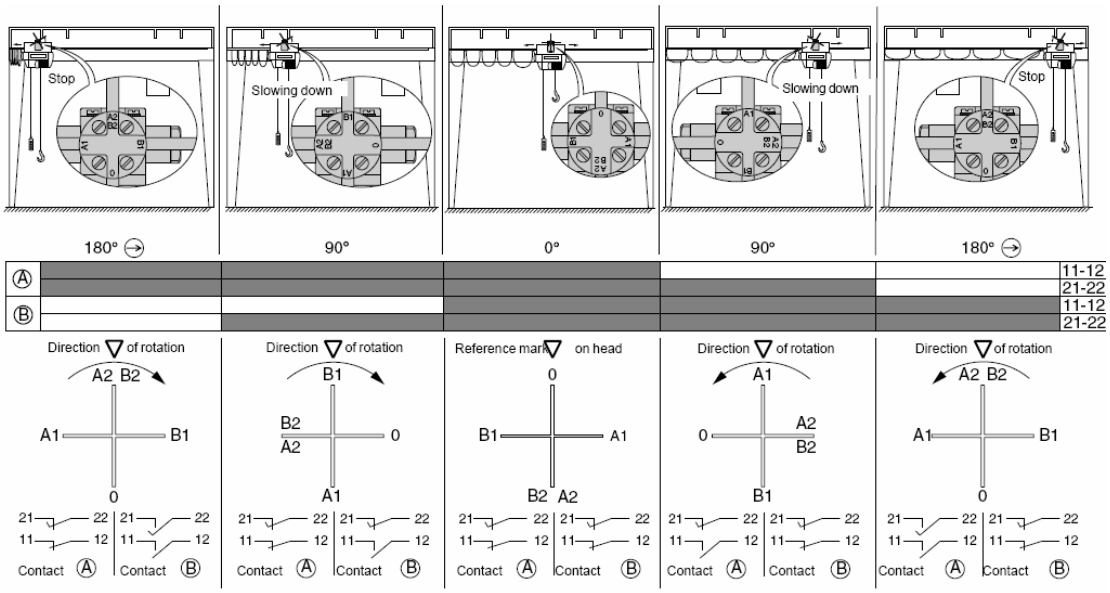
Altivar 71 **ATV71H075N4** 3-phase 400 Vac, 0.75 kW		 Terminal / Function ⏚ — Protective ground connection terminal R/L1, S/L2, T/L3 — Power supply PO — DC bus + polarity PA/+ — Output to braking resistor (+ polarity) PB — Output to braking resistor PC/- — DC bus - polarity U/T1, V/T2, W/T3 — Outputs to the motor
Altivar 71 **RS422 Encoder Interface Card** **VW3A3401**		
Harmony **Tower light** **XVB-C** **Green XVBC2B3** **Red XVBC2B4** **Yellow XVBC2B8** **Blue XVBC2B6**		
Signal Lamps and Illuminated Pushbutton **Harmony Style 5** **XB5AVB1** **XB5AW36B5**		

Limit Switch

OsiSense



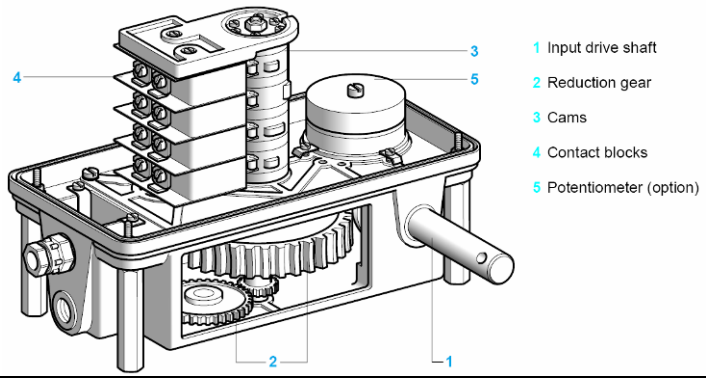
2 x 2-pole N/C + N/C
break before make contacts
(non interchangeable contacts)

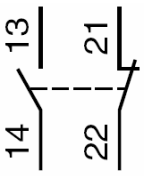
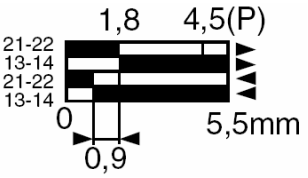
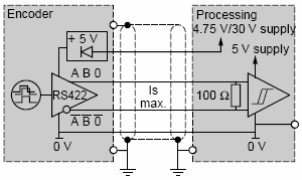
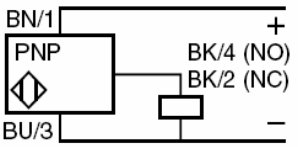


Screw Limit Switch
(3rd Party Component)

TER International

Reference depends on the Scaling rates



Overload Limit Switch **OsiSense** **XCKM110**		 
Incremental Encoder **OsiSense** **XCC1510PS11R** Spring coupling **XCCRAR1010**	 	*type R (N): 5 V output driver, RS 422, 4.5…5.5 V.* 
Inductive Proximity Sensor **OsiSense** **XS618B1PBL2**		 **BU : Blue** **BN : Brown** **BK : Black**

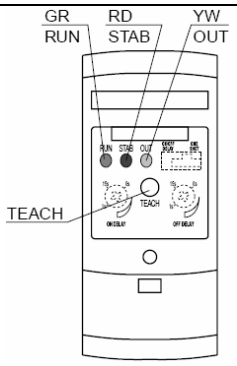
<table>
<tr>
<td>Photo-electric Sensor

OsiSense

XUX1ARCNT16

REFLECTOR

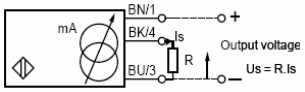
XUZC80</td>
<td> 

  </td>
<td> 

Relay output :
1 ⊘ ≂
2 ⊘ ≂
3 ⊘ NO
4 ⊘ Relay common
5 ⊘ NC
Transmitter:
1 ⊘ ≂
2 ⊘ ≂</td>
</tr>
<tr>
<td>Inductive Proximity Sensor

OsiSense

XS9C111A1M12</td>
<td>  </td>
<td>M12
 

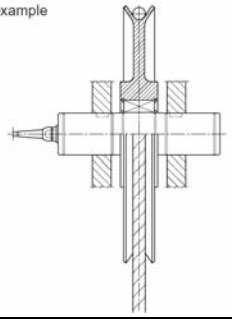
<table>
<tr><th>Output voltage</th><th>Load impedance value</th></tr>
<tr><td>0…10 V</td><td>R = 1000 Ω</td></tr>
</table></td>
</tr>
<tr>
<td>Load Cell (3rd Party Component)

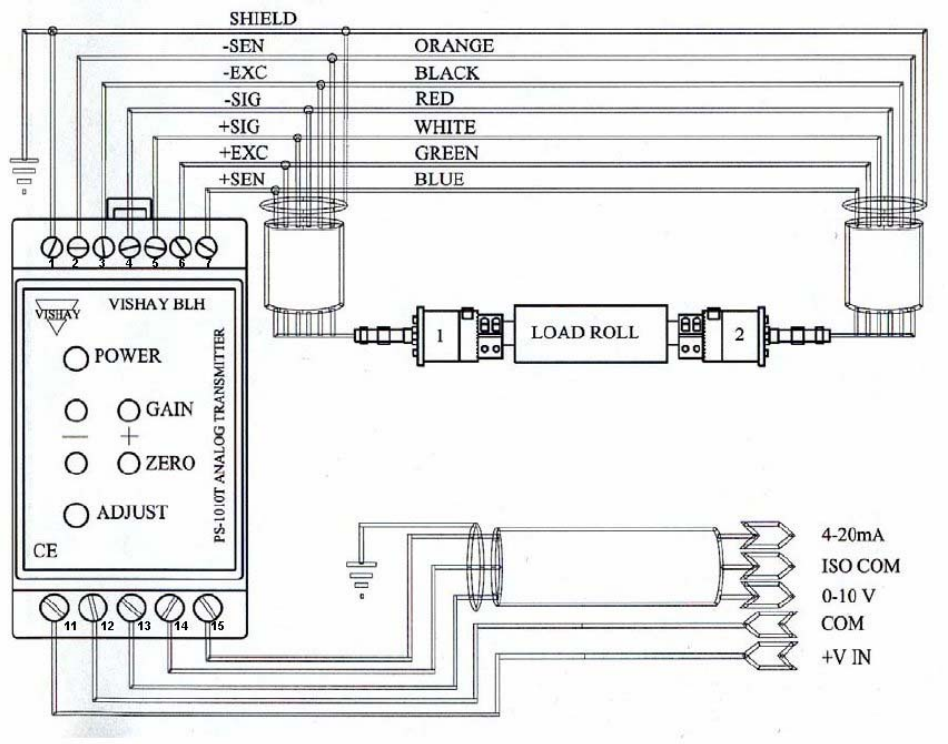
Vishay

KISD-6

Web Tension Transmitter

PS-1010T</td>
<td> 

Installation example
  </td>
<td>  </td>
</tr>
</table>



Amplifier Inputs:

1. Shield
2. –SEN (Orange)
3. –EXC (Black)
4. –SIG (Red)
5. +SIG (White)
6. +EXC (Green)
7. +SEN (Blue)

Amplifier Outputs:

8. +VIN
9. COM
10. 0-10V
11. ISOCOM
12. 4-20mA

Note:

1. For this application that uses a four-wire load cell cable, two jumpers must be installed between +SEN and +EXC and between –SEN and –EXC.

2. The 4 to 20 mA outputs are used for this architecture.

Software

General

The main programming work is the programming of the Modicon M238 Logic controller, the configuration of the CANopen fieldbus and creating the screens for the HMI display if it's used.

Programming the M238 is done by using SoMachine.

Programming of the Magelis XBTGT2330 HMI is done by using Vijeo Designer, which is integrated in the SoMachine software tool.

Configuration of the drives (ATV312 and ATV71) is done using the control panel on the drive.

To use the software packages, your PC must have the appropriate Microsoft Windows operating system installed:

- Windows XP Professional

The software tools have the following default install paths:

- SoMachine
  C:\Program Files\Schneider Electric\SoMachine

- Vijeo Designer (Installed with SoMachine)
  C:\Program Files\Schneider Electric\Vijeo Designer

Software Libraries & Function Blocks

Schneider Electric offers a hoisting relevant function block library.

Hoisting.lib includes the following function blocks:

- Anti-crab
- Limit switch management
- Load overspeed control
- Overload control
- Smooth slewing
- Speed select
- Speed optimization and rope slack
- Hoisting position synchronization
- Wind speed control
- Scale Input
- Monitoring data storage
- Anti-sway open loop

All function blocks are described in detail in their separate user guides.

The following function blocks are used in the Hoisting_Optimized_CANopen_M238 solution:

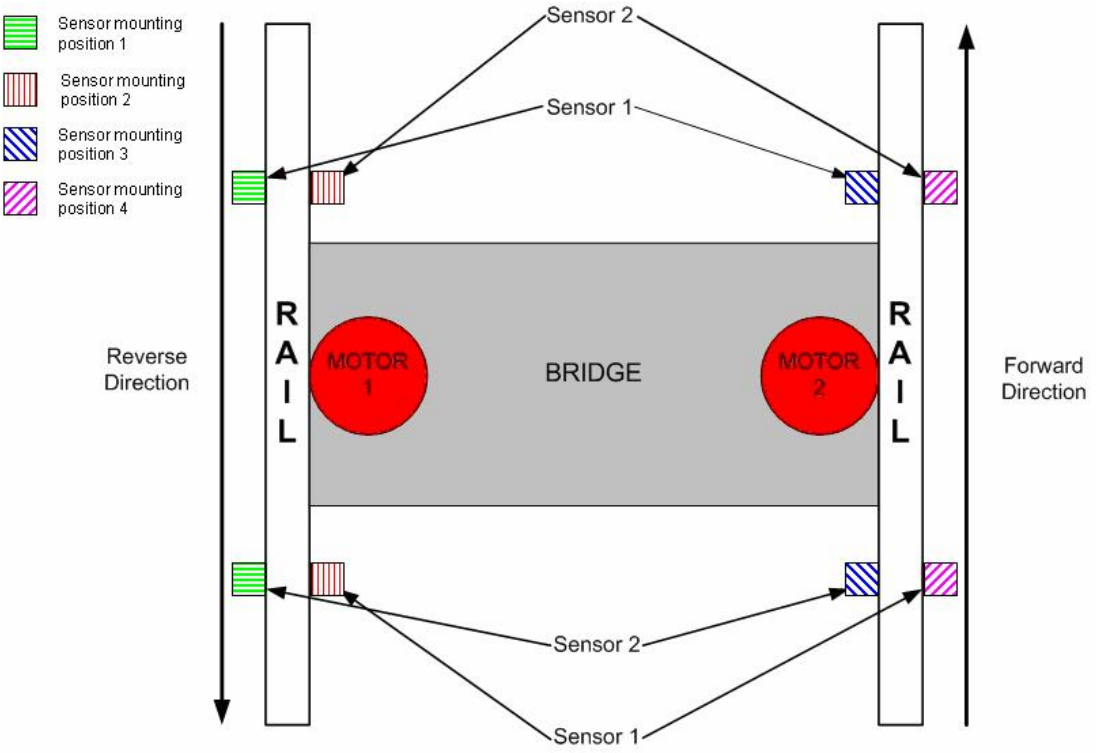
- Anti-crab
- Antisway
- Limit switch management
- Load overspeed control
- Overload control
- Speed optimization and rope slack
- Hoisting position synchronization
- Speed select
- Scale input
- Wind speed control
- Altivar31_Control_FB
- Altivar71_Control_FB

Note: The Altivar31_Control_FB, Altivar71_Control_FB, are part of the **Altivar Library**.

More detailed information on the function blocks can be found in the documentation of the Hoisting library and in the function block user guides.

Anti-crab

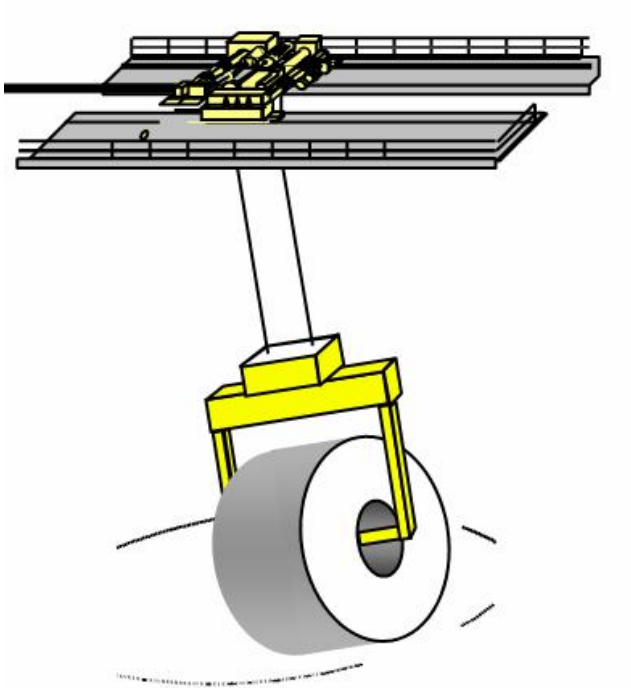
The Anti-crab function detects a skew or drifts in the bridge against the rail and calculates a corrected speed for both of the drives in order to maintain the bridge parallel to the rail.



Anti-sway

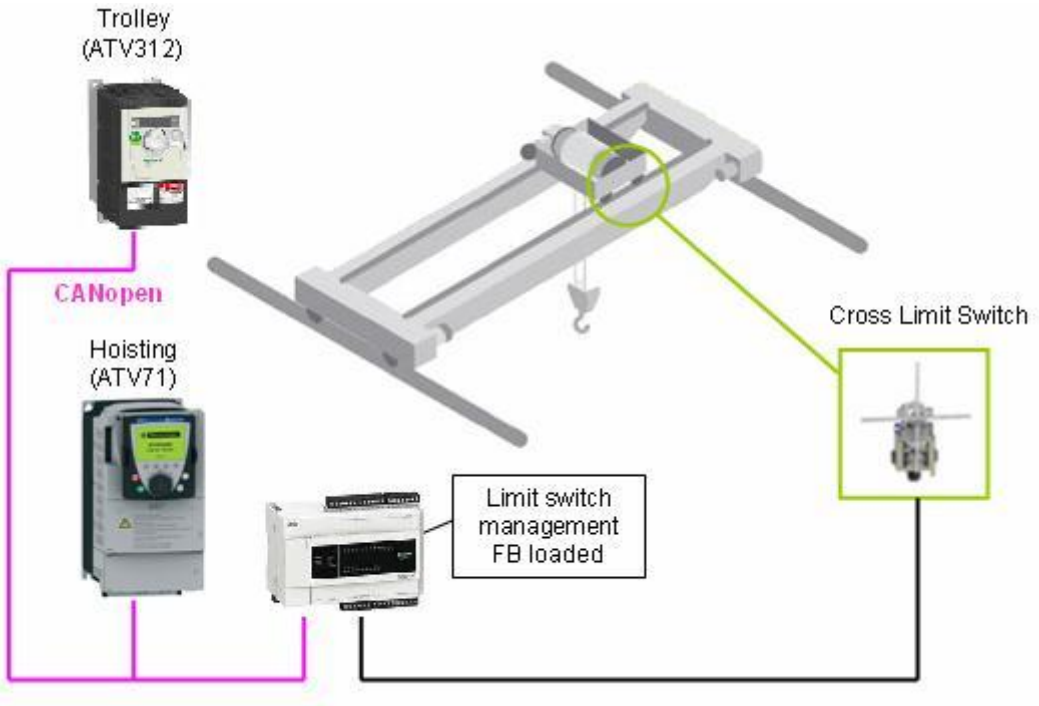
The Anti-sway function interacts with one axis (trolley or travel) and, based on the operational input, calculates a movement profile so that the drive compensates for the normal sway effect of the load.

The Anti-sway solution from Schneider Electric works without using sensors to measure the amount of sway.



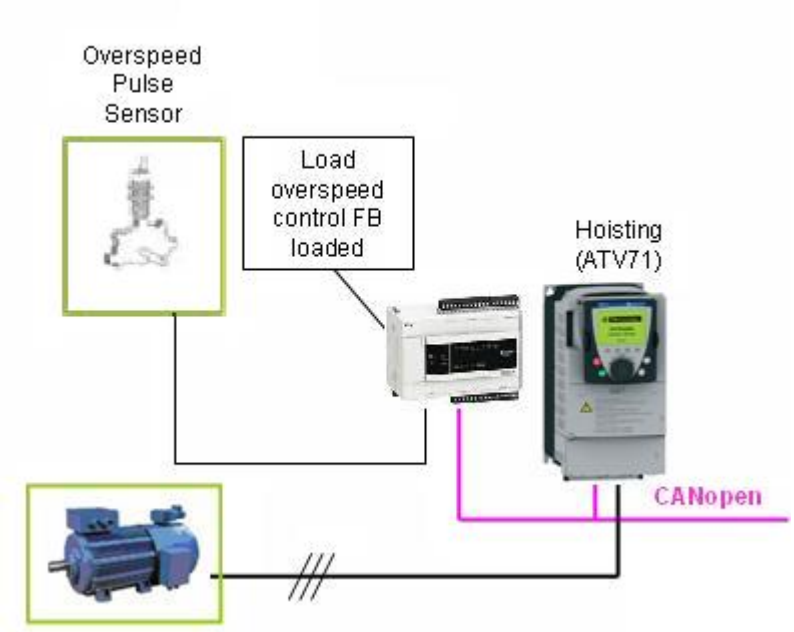
Limit switch management

The Limit switch management function block limits the movement of the Trolley/Bridge/Hoist along a rail. This function block controls the Trolley/Bridge/Hoist by stopping/slowing it down according to the status of limit switch/sensors.



Load overspeed control

The Load overspeed control function block has two functions. It detects if the hoist drive can hold the load during the hoist movement. This is done by checking the actual speed against the maximum speed via the pulse sensor. While the drive is stopped and the brake is on, it detects load movement due to brake failure.



Monitoring data storage

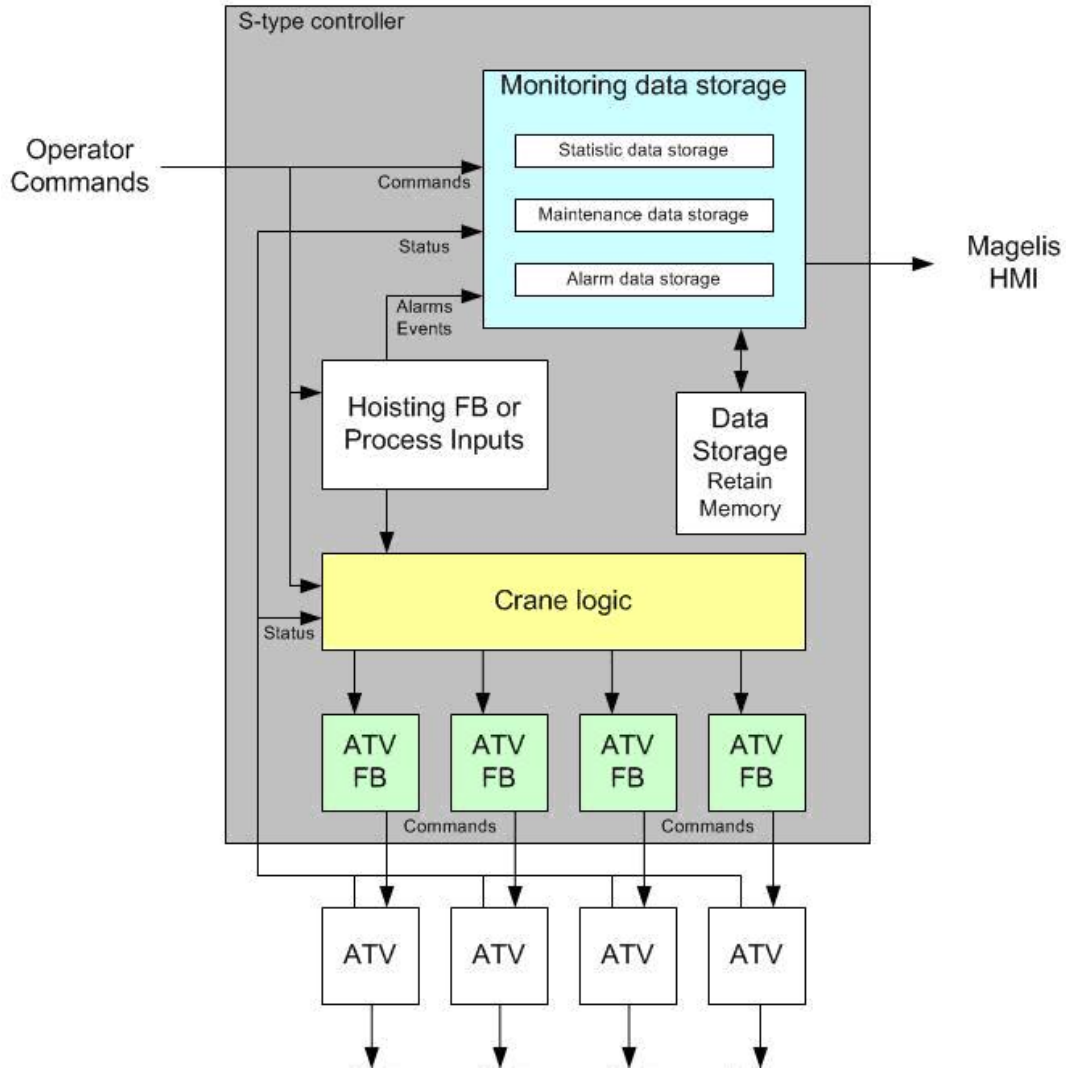
The Monitoring data storage function consists of three parts.

- Alarm data storage function block series
- Statistic data storage function block
- Maintenance data storage function block.

The Alarm data storage function block series are pre-configured for the following events: Overload, Overspeed, Encoder Alarm, Overtorque and Load current. All event signals are archived with their respective date, time and duration.

The Statistic data storage function block records all events that are part of the movement. The function block records four separate movements. Hoisting, Trolley, Bridge travel and Slewing. Depending on the crane, the inputs can also be used for other movements. For each type of movement the function block counts the events and calculates an overall runtime for each axis.

The maintenance block uses actual loading information to calculate the amount of used ‘theoretical lifetime’ for each movement and generates an alarm if maintenance is needed.



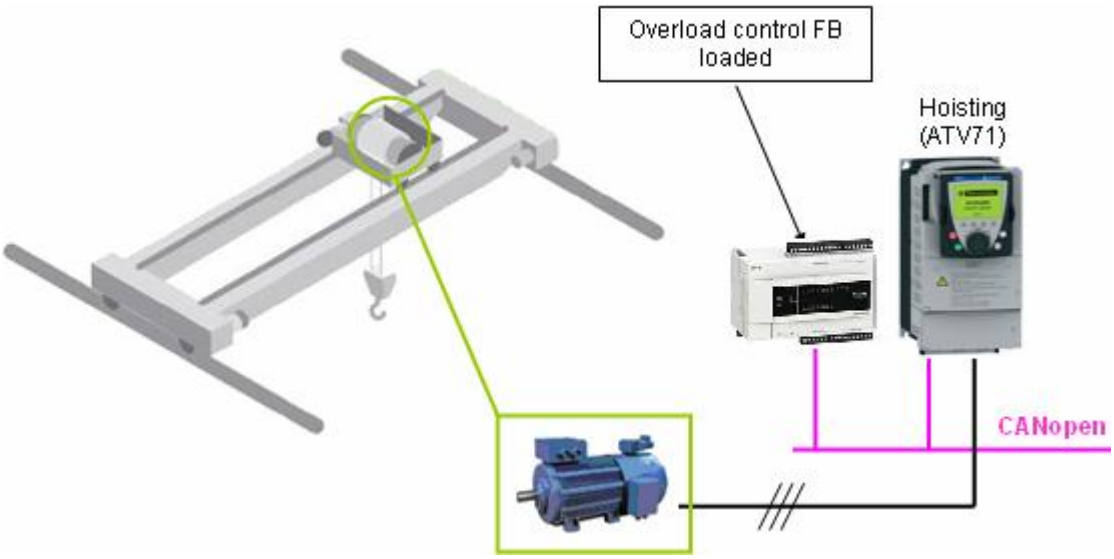
Overload control

The Overload control function block use the actual torque and evaluates it against a maximum torque parameter. The overload alarm can be reset in three different ways. For each method, the Schneider Electric hoisting library includes a separate function block.

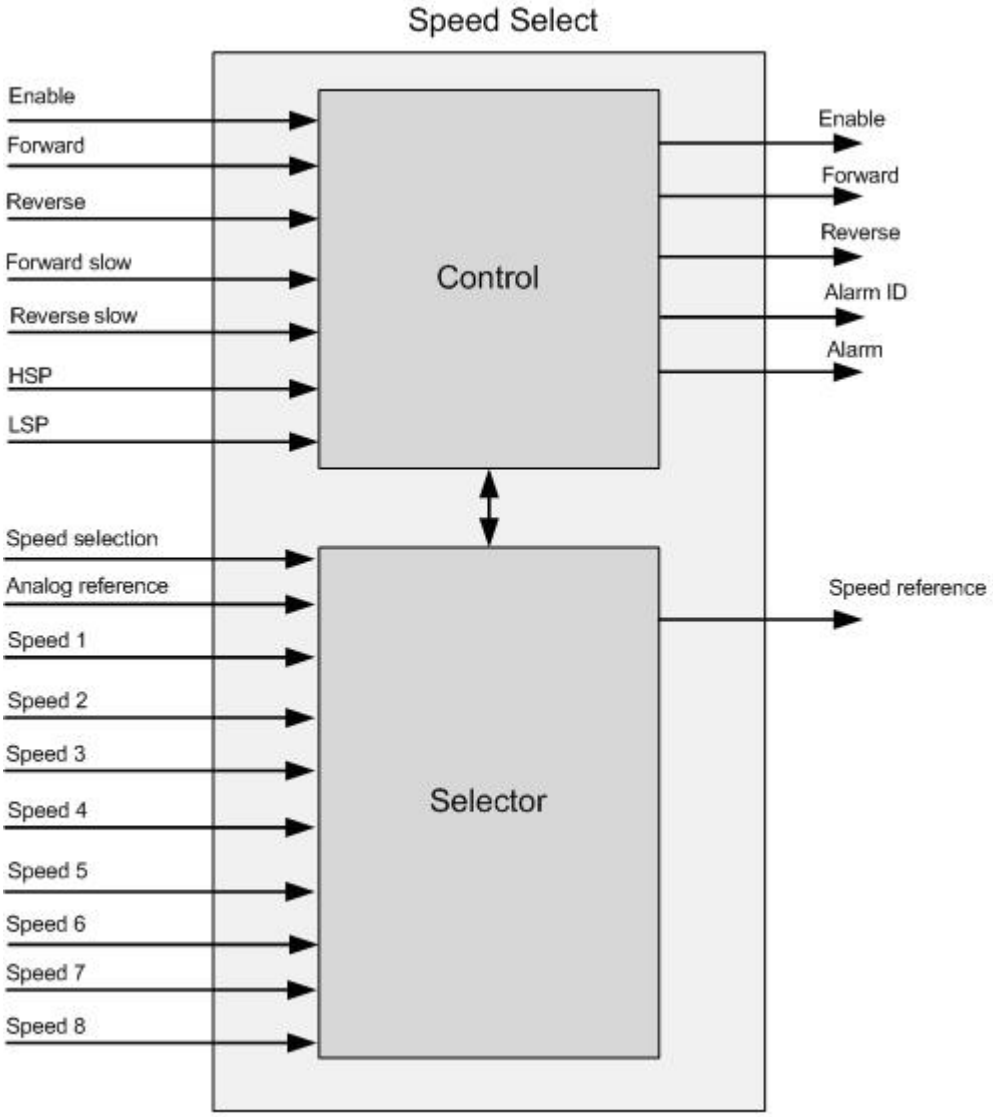
The Overload Torque method can be reset by returning the load to a configured hook torque value.

In the event of an overload, using the actual RPM and time, the Overload Distance method calculates the distance and position until the upward movement stops. Using the same method the function block calculates the distance to the original starting point.
To reset an overload alarm, the actual torque must drop below the maximum torque value and the calculated position must be equal to or lower than the position when the overload event occurred

The Overload Encoder method stores the encoder value if an alarm is recognized and resets it after moving back to the stored encoder value and the torque value is less than the maximum torque.

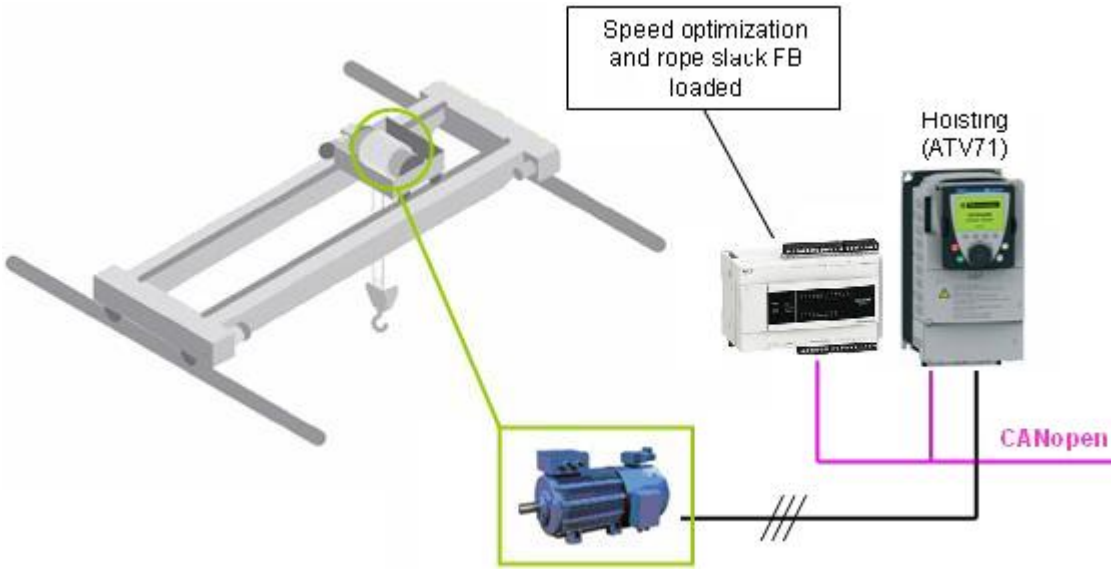


Speed select The Speed select function block is used to select a speed value from different sources. The required speed value can be either a fixed or an analog value.

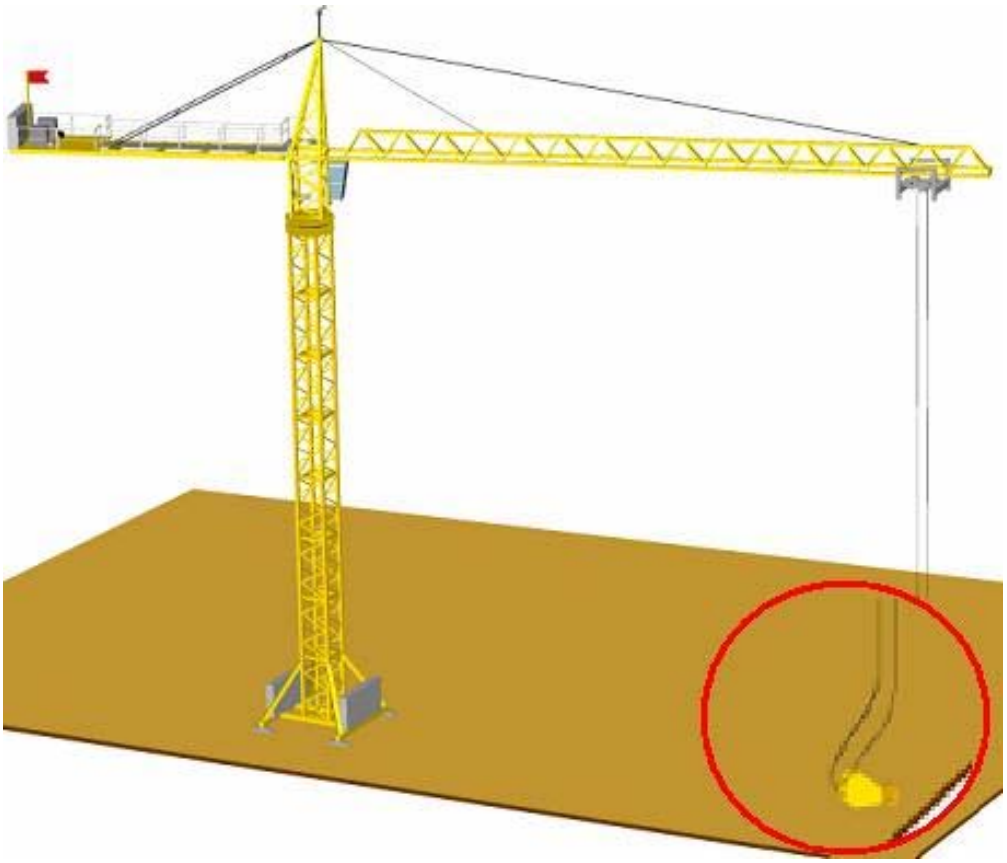


Speed optimization & rope slack

The speed optimization function gives the operator the possibility to use a higher speed value as the optimum speed for the hoisting movement with the actual load. The actual speed is compared with the actual and maximum allowable torque to obtain the optimum speed for the current load.



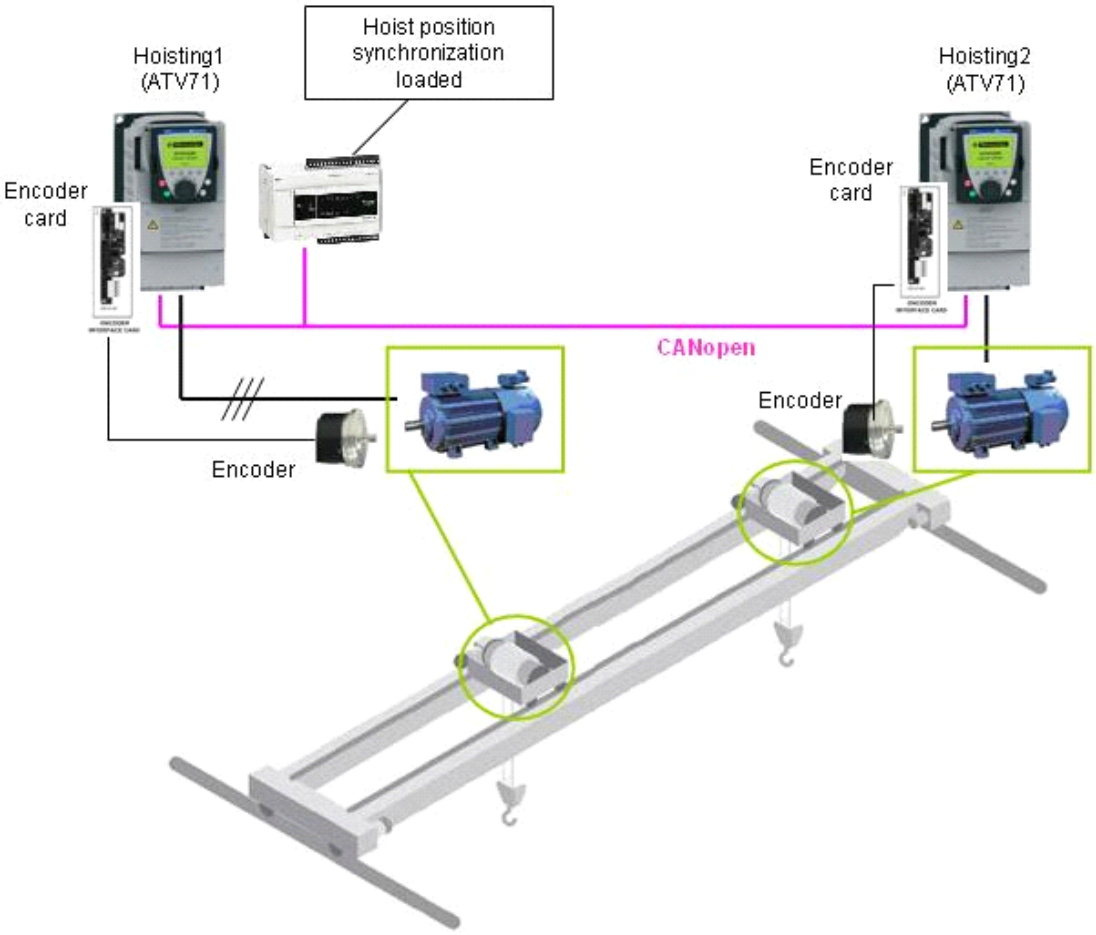
The Rope slack function detects when the load drops below a minimum torque (the weight of the hook) and reduces the speed to the minimum LSP speed. This protects the crane against unwinding the whole cable from the drum.



Hoisting position synchronization

The Hoisting position synchronization function for industrial cranes is designed to synchronize the movement of two hoists in two trolleys for simultaneous movement.

The image below shows an over-head crane with a Hoisting position synchronization function.



Communication

General

The TVDA architecture includes two different communication networks.

The CANopen fieldbus connects the Modicon M238 Logic contoller as CANopen Master and Altivar drives as CANopen slaves. All the drives are connected via CANopen TAPs. The CANopen transmission rate is 500 kb/s.

The M238 and the HMI communicate using the SoMachine protocol.

The download from the PC to M238 and to the HMI is done using a single cable connection. The PC has to be connected to the HMI over USB. Using this connection the data is also sent across to the M238.

For hardware communication:

- CANopen fieldbus: M238 ⇔ ATV71 and ATV312
- SoMachine protocol: M238 ⇔ XBTGT

For programming:

- SoMachine protocol PC ⇔ XBTGT and M238

The local control panel is used to configure the ATV312 and the ATV71.

PC ⇔ XBTGT ⇔ M238

The download direction is from the PC to the HMI and via the HMI to the controller.

1. PC
2. HMI XBTGT
3. Modicon M238
4. USB to USB cable **XBTZG935**
5. SubD9 to RJ45 cable **XBTZ9008**

Note:
For a direct connection from the PC to the controller the **TCSXCNAMUM3P** cable should be used.

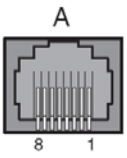
M238 ↔ HMI **XBTZ9008** Cable for connecting XBTGT and M238	
PC ↔ HMI PC connection cable **XBTZG935** Cable for the connection between a SoMachine-equipped PC and XBTGT	

CANopen

ATV71 and ATV312

Modbus / CANopen port

Note:
In case of CANopen, the CANopen tap is used to connect the drive to the **CANopen** bus via RJ45.



Pin	Signal	Description
1	CAN_H	data wire
2	CAN_L	data wire, inverted
7	MOD+10V_OUT	10V power supply
8	MOD_0V	Reference potential for MOD+10V_OUT

Pin	Signal	Description	
4	MOD_D1	Bidirectional transmit/receive signal	RS485 level
5	MOD_D0	Bidirectional transmit/receive signal, inverted	RS485 level
7	MOD+10V_OUT	10 V power supply, max. 150 mA	Output
8	MOD_0V	Reference potential for MOD+10V_OUT	Output

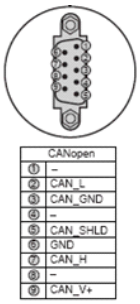
CANopen tap

TSXCANTDM4

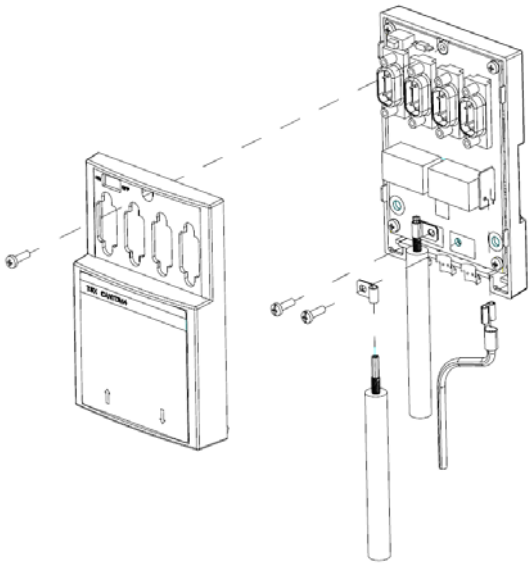
4 port CANopen tap

For the purpose of this application, the sliding switch should be set to OFF if it is not at the end of the CANopen line.

At the end of the bus, the terminating resistor must be active. To do this, set the switch on the tab to ON. The bus cable must be connected to the in-coming side.



CANopen	
①	–
②	CAN_L
③	CAN_GND
④	–
⑤	CAN_SHLD
⑥	GND
⑦	CAN_H
⑧	–
⑨	CAN_V+

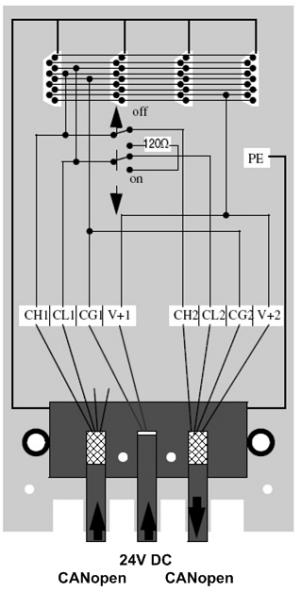


CANopen tap

TSXCANTDM4

Power supply:

V+1 24 Vdc
CG1 0 Vdc



Signal	Terminal block 1	Terminal block 2	Wire color
CAN_H	CH1	CH2	white
CAN_L	CL1	CL2	blue
CAN_GND	CG1	CG2	black
CAN_V+	V+1	V+2	red

CANopen pre-assembled connection cable

TCSCCN4F3M3T

This cable is used to connect the CANopen tap to the ATV312 and ATV71.

CANopen cable

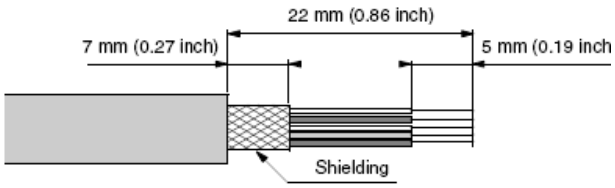
TSXCANCxy

The cable is available in various versions (x):

- A - Standard
- B - No Flame
- D - Heavy Duty

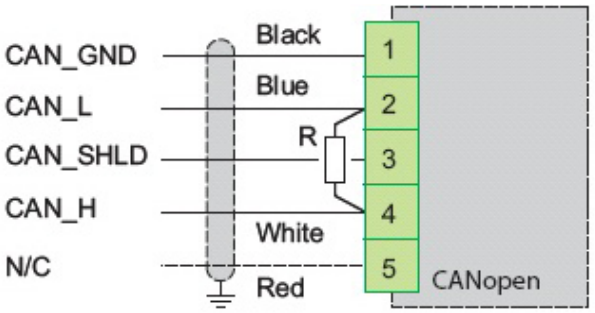
and various lengths (y):

- 50 - for 50 m
- 100 - for 100 m,
- 300 - for 300 m.



M238 CANopen

If the M238 is installed at the beginning of the CANopen you have to install a terminating resistor (120 Ohm) between **terminal 2 CAN_L** and **terminal 4 CAN_H.**



Implementation

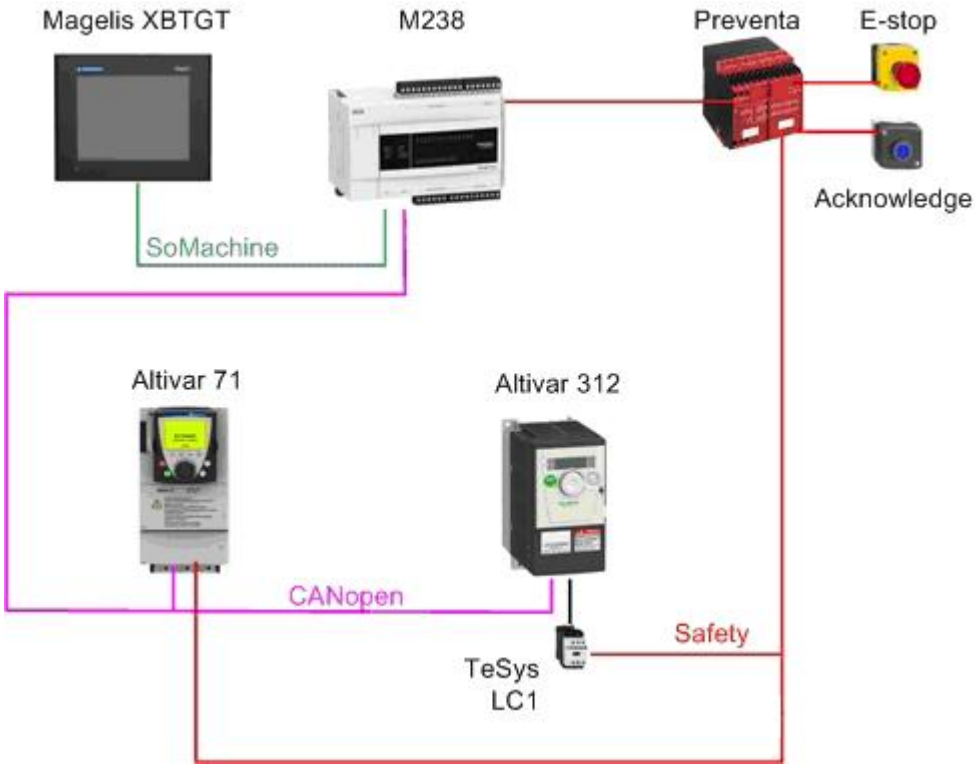
Introduction

This chapter describes all the steps necessary to initialize, configure, and program the system to achieve the described application functions.

Although the implementation of the application source code for the controller and the devices includes a combination of function blocks which fulfill several application and device functions for the customer, this solution is not a complete user program for a crane.

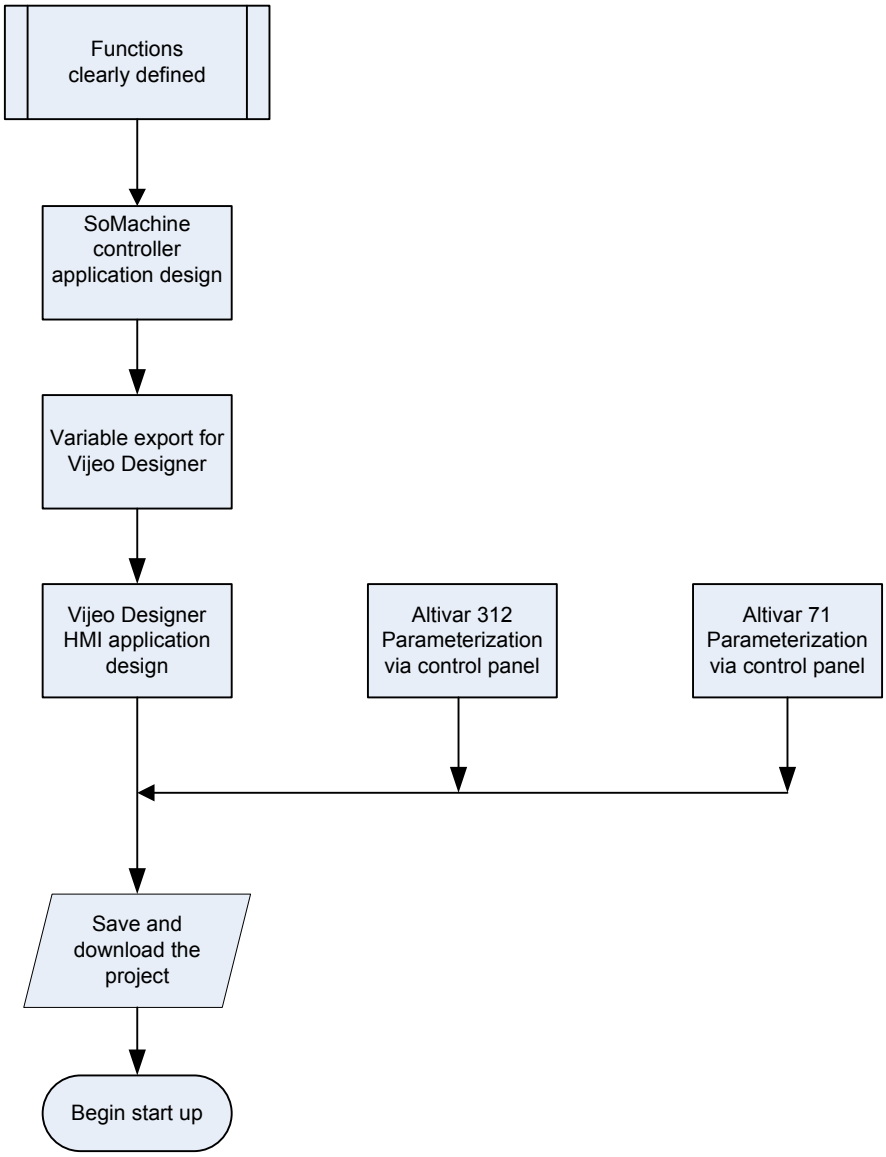
The Application Function Block Library, together with the Schneider Electric device function blocks (for example, for drives), facilitate a full adaptation for the OEM. The following chapter describes, step by step, how to implement the hoisting FBs in the user program. The AFB blocks have been tested and validated in a structured program. The Application Design Experts, together with the OEM must adapt them to fulfill the exact requirements for a complete crane application.

Functional Layout



Course of Action

Flow chart of the implementation procedure:



Communication

Introduction

This chapter describes the data passed via the communication fieldbus (example CANopen) that is not bound directly with digital or analog hardware.
The list contains:

- The device links
- Direction of data flow
- Symbolic name and
- Bus address of the device concerned.

Device Links

CANopen fieldbus systems are used in this application for the communication with the drives.

SoMachine protocol is used to communicate with the HMI.

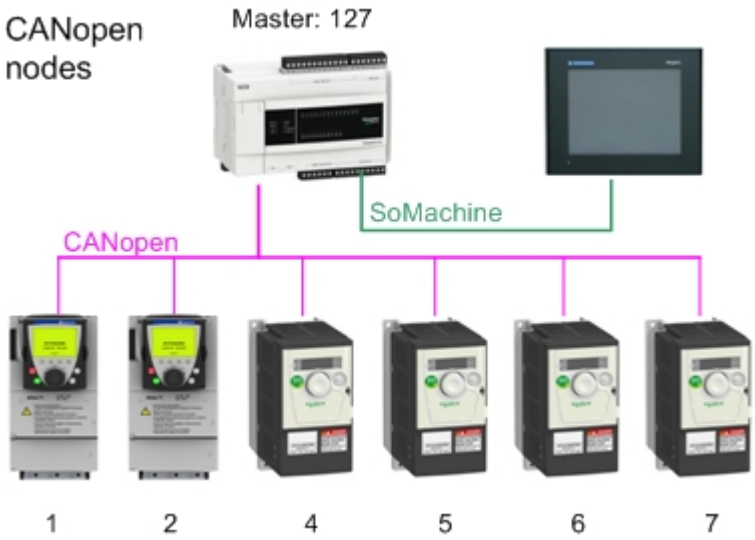
The following devices are networked via **CANopen**:

- Modicon M238 Logic controller, bus address 127 (master)
- 4 Altivar 312 variable speed drives, bus address 4, 5, 6 and 7
- 2 Altivar 71 variable speed drives, bus address 1 and 2

CANopen Transmission rate 500 kb/s.

Communication

CANopen & SoMachine



NOTE

CANopen device addresses (Node IDs) are given in decimal while all COB-IDs and CANopen object addresses are in hexadecimal.

CANopen Addresses

M238 is *CANopen master*	
Device	**CANopen Address**
M238	127
ATV71	1
ATV71	2
ATV312	4
ATV312	5
ATV312	6
ATV312	7

Datalink M238 ⇔ ATV71

ATV71		
Data Direction Device ➔ M238 (SPDO)		
Mapping	**Index /Subindex**	**Designation**
Send PDO1	6041	Drivecom status register
	6044	Control effort
	2002/6	Motor torque
	2002/5	Motor current
Send PDO2	2016/3	IL1R
	201A/C	PUC

Data Direction M238 ➔ Device (RPDO)		
Mapping	**Index /Subindex**	**Designation**
Receive PDO1	6040	Drivecom command register
	6042	Target velocity
	203C/2	Acceleration
	203C/3	Deceleration

Datalink M238 ⇔ ATV312

ATV312		
Data Direction Device ➔ M238 (SPDO)		
Mapping	**Index /Subindex**	**Designation**
Send PDO1	6041	Drivecom status register
Send PDO6	2016/29	Logic input/output image
	6044	Control effort
	2002/05	Motor current
	2016/2D	Physical Value AI3

Data Direction M238 ➔ Device (RPDO)		
Mapping	**Index /Subindex**	**Designation**
Receive PDO1	6040	Drivecom command register
Receive PDO6	6042	Target velocity
	203C/2	Acceleration ramp time
	203C/3	Deceleration ramp time

Controller

Introduction

The controller chapter describes the steps required for the initialization and configuration and the source program required to fulfill the functions.

Requirements

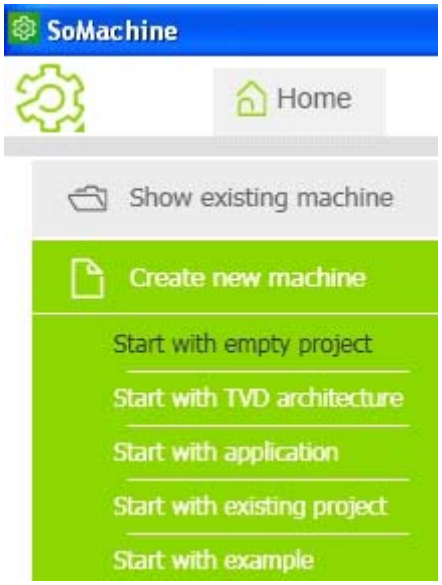
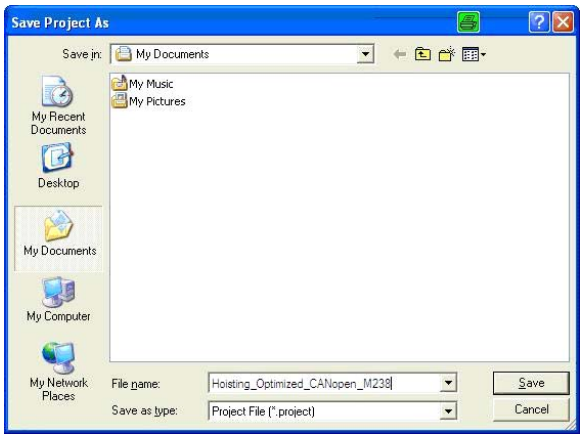
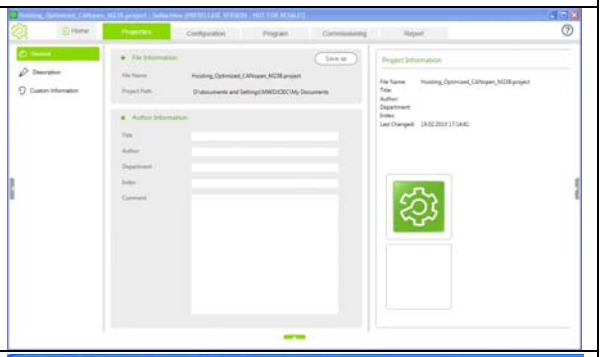
The following is required before proceeding with the controller configuration:

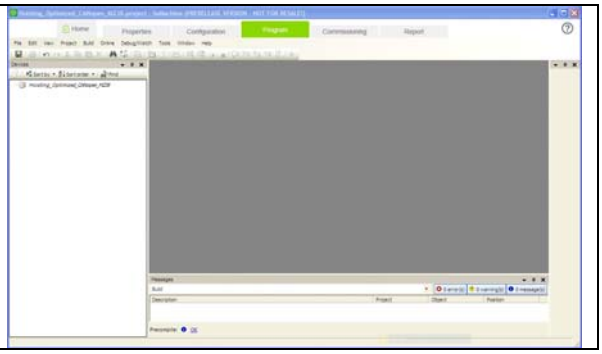
- SoMachine is installed on your PC
- The Modicon M238 Logic controller is switched on and running
- The M238 is connected to the HMI with the programming cable XBTZ9008 (M238 to HMI)
- The HMI is connected to the PC via the cable XBTZG935 (HMI to PC)

Setting up the Controller is done as follows:

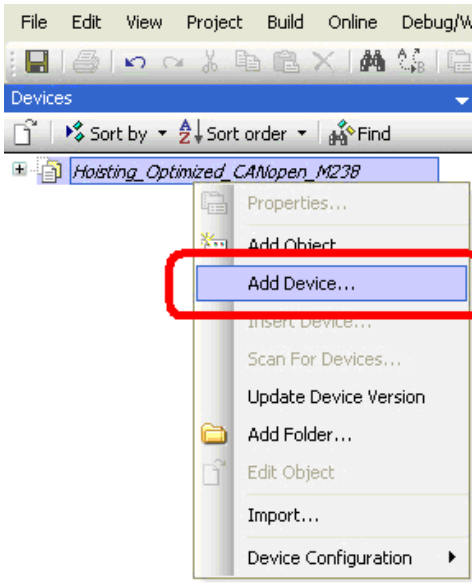
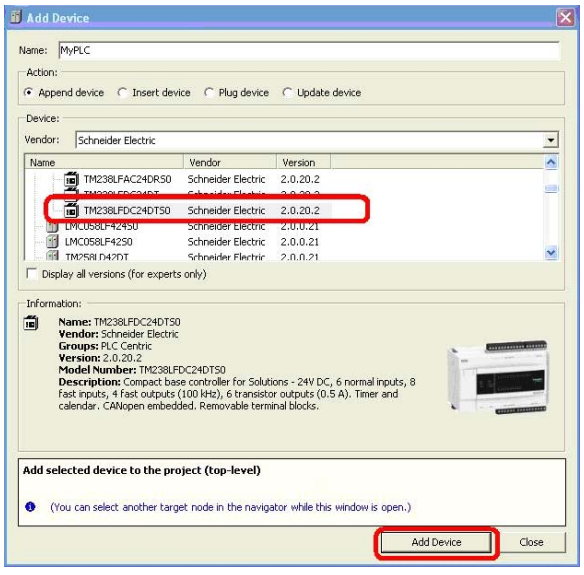
- Create a new project
- Add the Controller
- Add the CANopen fieldbus
- Add CANopen devices
- ATV71 CANopen configuration
- ATV312 CANopen configuration
- Hardware Layout
- Add Hoisting Library
- Add POU
- Task configuration
- Add Vijeo Designer HMI
- Configure Controller ↔ HMI Data Exchange
- Communication Setting Controller ↔ PC
- Communication Setting HMI ↔ PC
- Save the Project
- Build Application
- Download the Controller and HMI project
- Login to the Controller
- Application overview

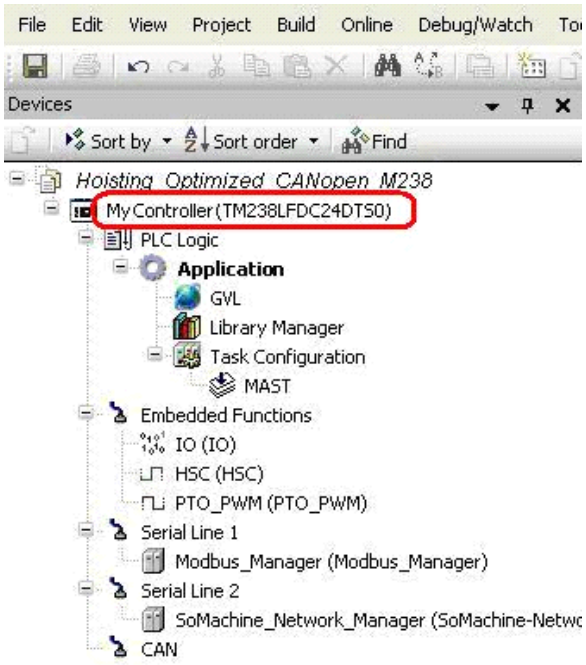
Create a new Project

1	To create a new project select **Create new machine** → **Start with empty project**	
2	In the **Save Project As** dialog enter a **File name** and press **Save.** **Note :** As default the project is saved under My Documents.	
3	The SoMachine User Interface opens.	
4	In the User Interface select the **Program** tab	

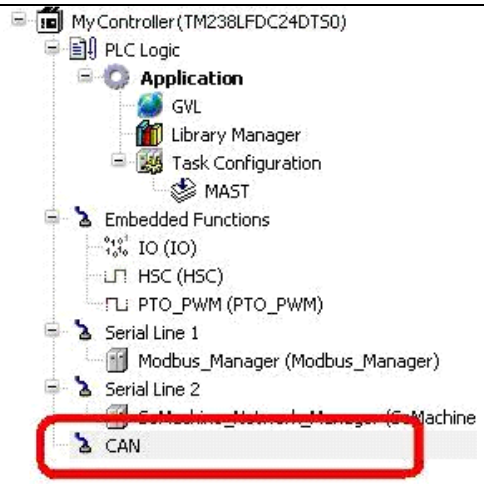
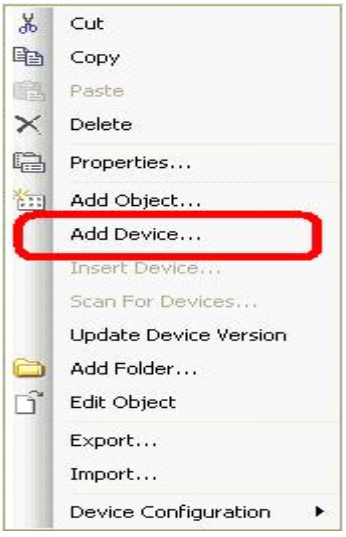
5	The Program window appears	

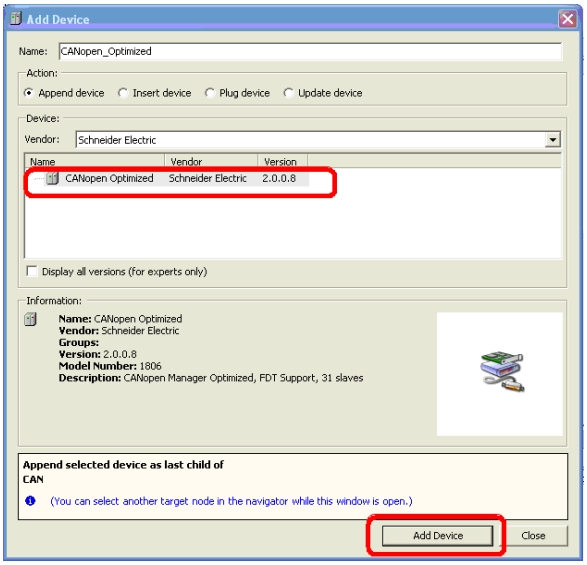
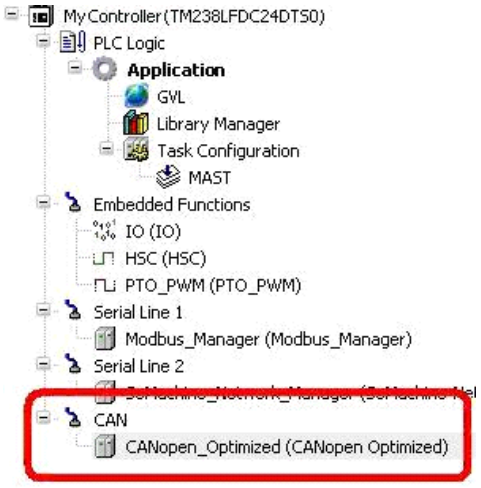
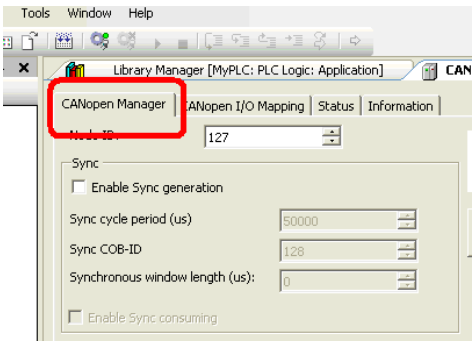
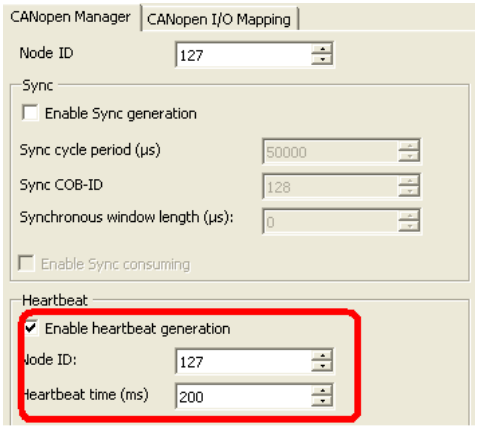
Add a Controller

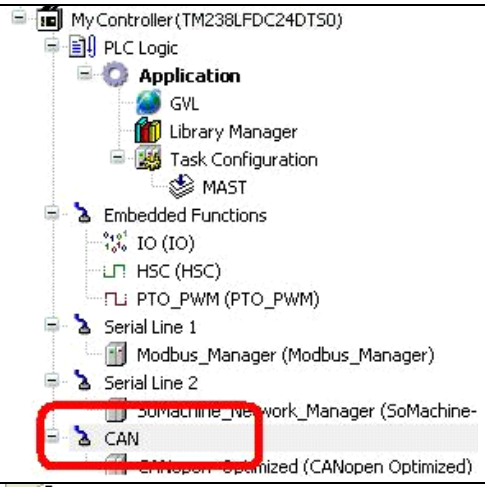
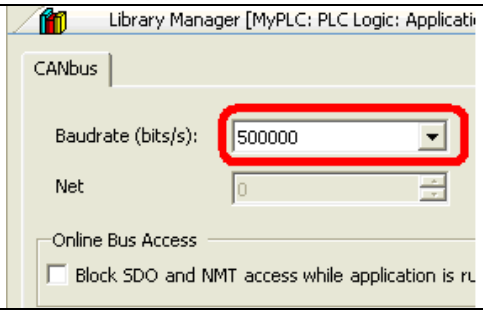
1	Right click on **Hoisting_Optimized_CANopen_M238** Click on **Add Device…**	
2	Select the controller **M238 (TM238LFDC24DTS0)** Click **Add Device**	

3	The **Devices** folder displays the new controller	

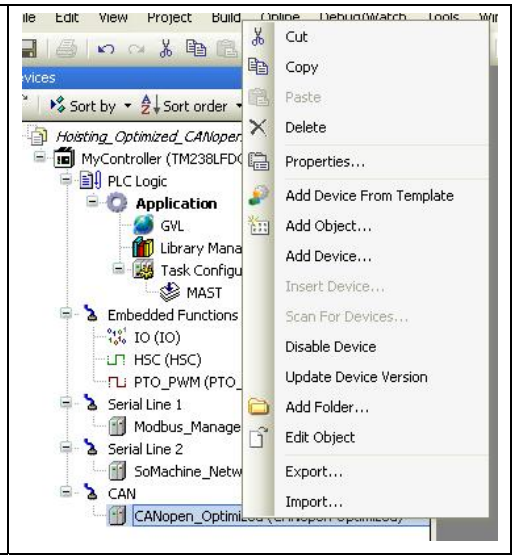
Add the CANopen fieldbus

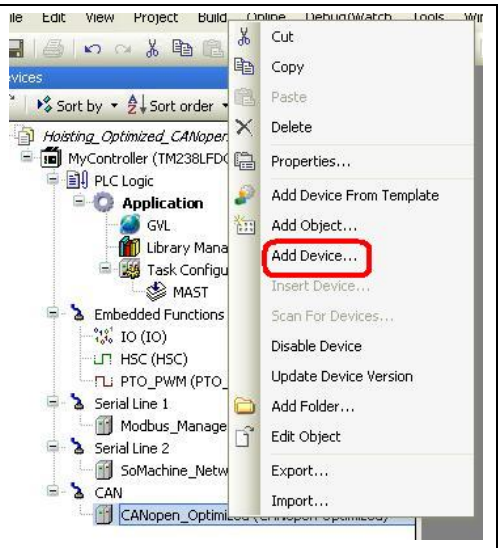
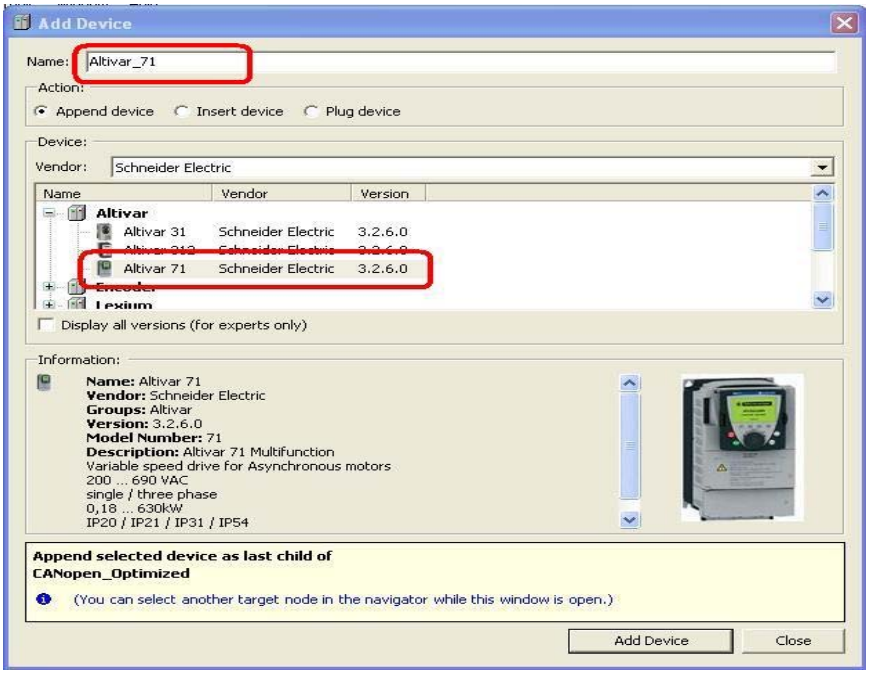
1	Right click on **CAN**	
2	Click on **Add Device…**	

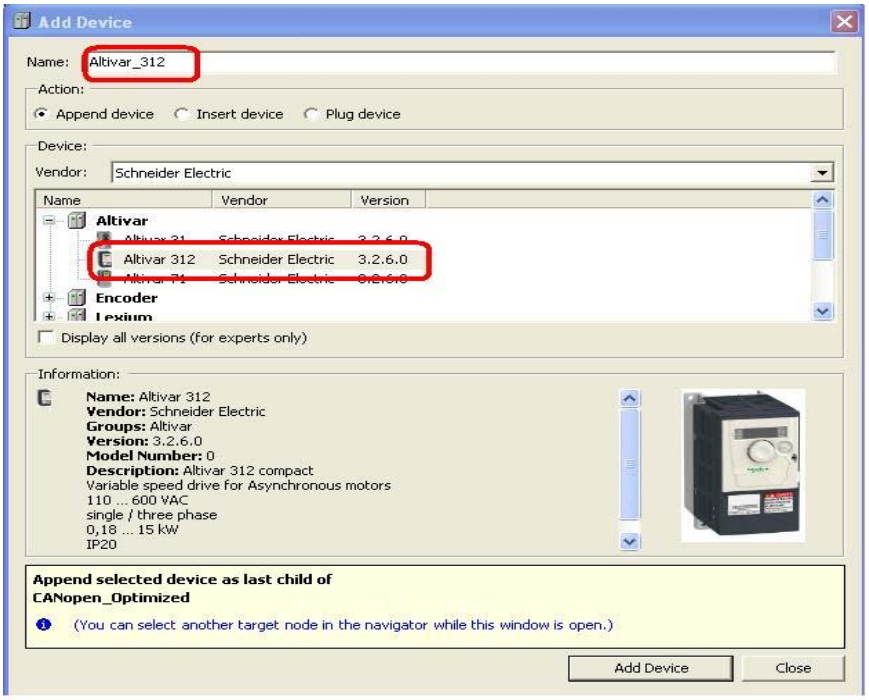
3	Select **CANopen Optimized** Click on **Add Device**	
4	To activate the Heartbeat of the M238 **double click** on **CANopen_Optimized.**	
5	Click on **CANopen Manager** tab	
6	Check **Enable heartbeat generation**. The **Heartbeat time** is **200** ms	

7	Set the **Baudrate** of the CANopen with **Double click** on **CAN**	
8	Select **500000** (bits/s) as Baudrate	

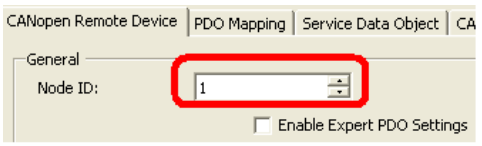
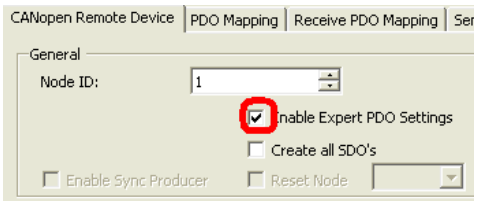
Add CANopen Devices

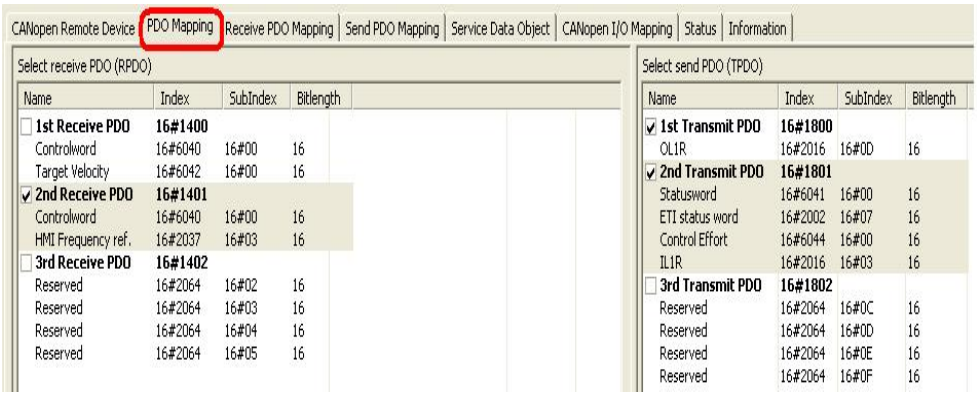
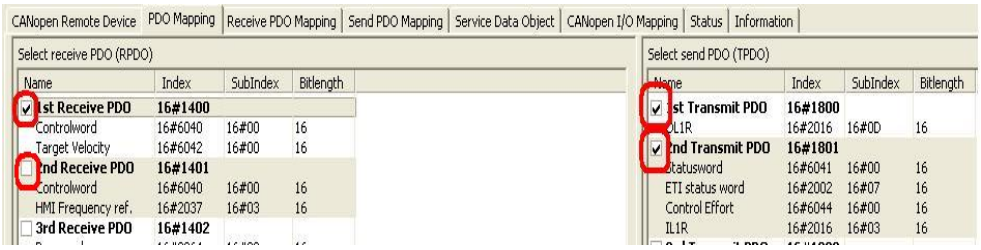
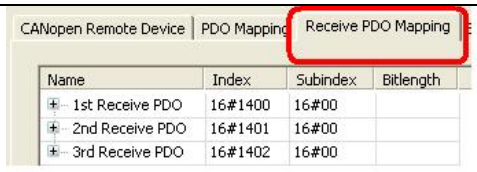
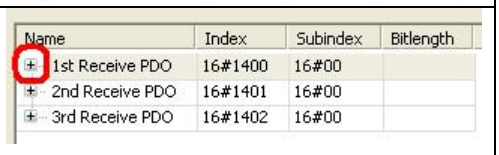
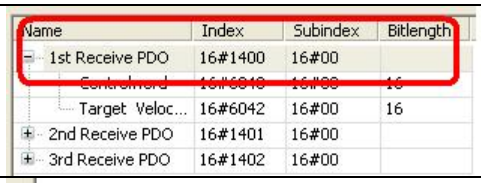
1	Right click on the **CANopen_Optimized** Select **Add Device…**	

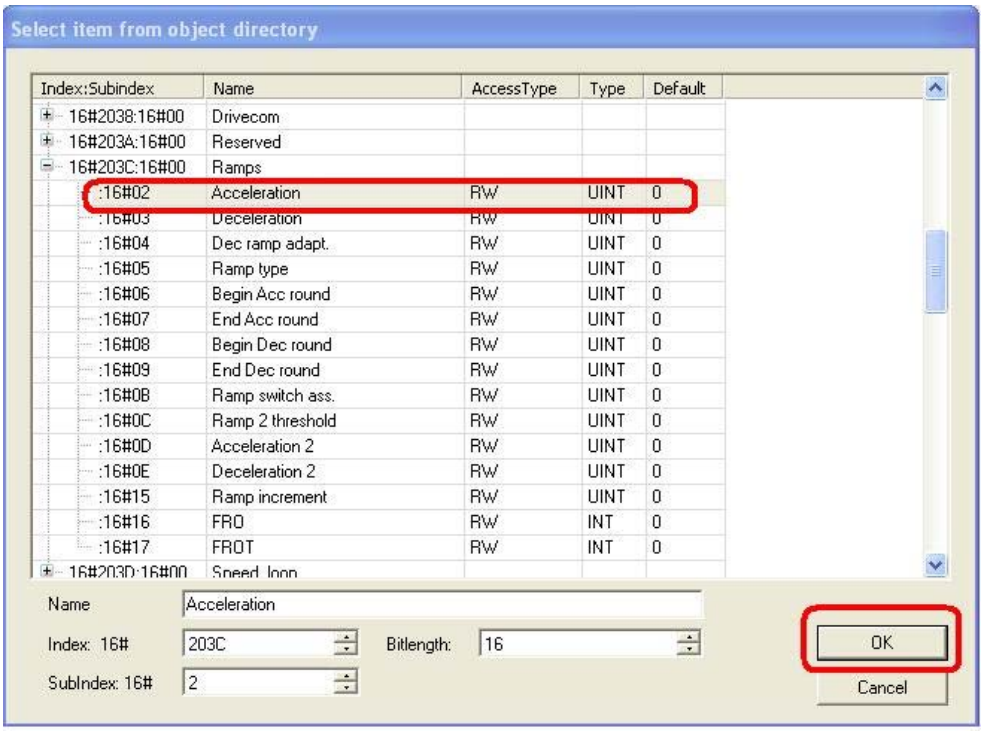
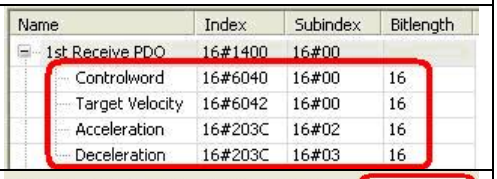
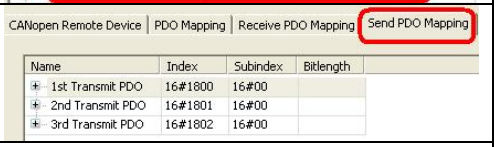
2	Select **Add Device…**	
3	Select the following devices are connected to the CANopen fieldbus: **2 x ATV71** and **4 x ATV312** Add each device by clicking on **Add Device…**. Once you have added all devices click on **Close**. **Note:** The name of the device can be changed under Name. 	
	Note: The name of the device can be changed under **Name**.	

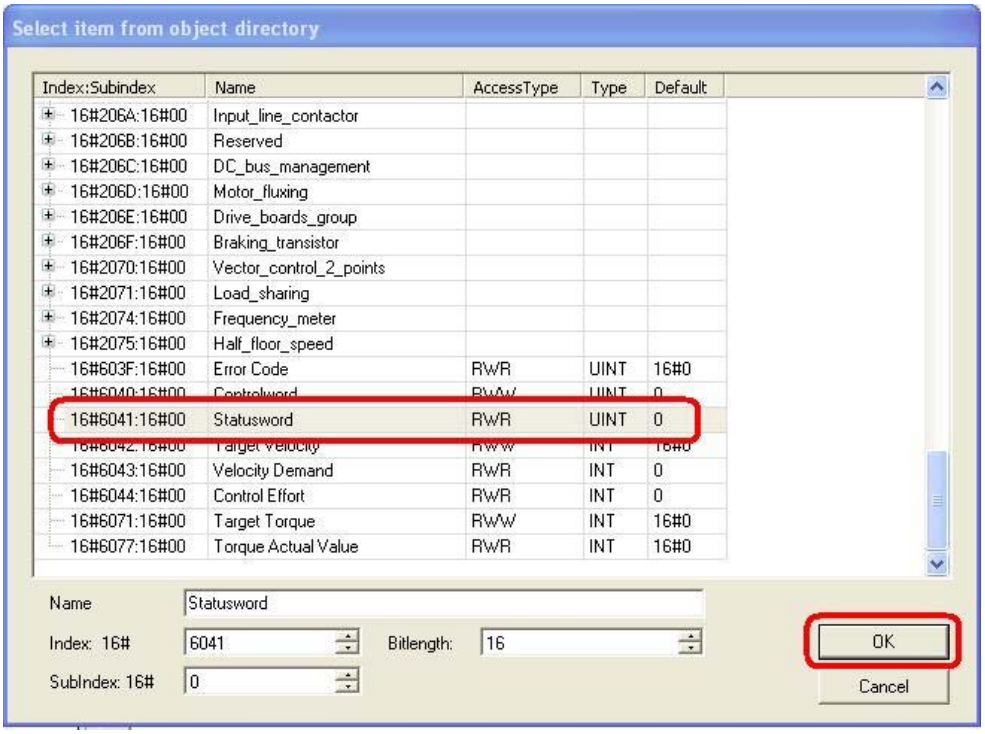
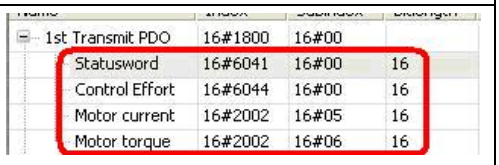
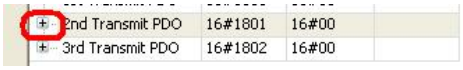


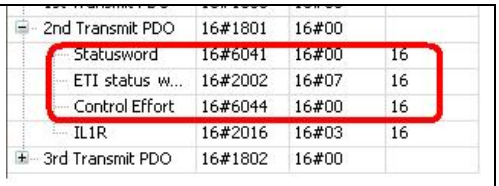
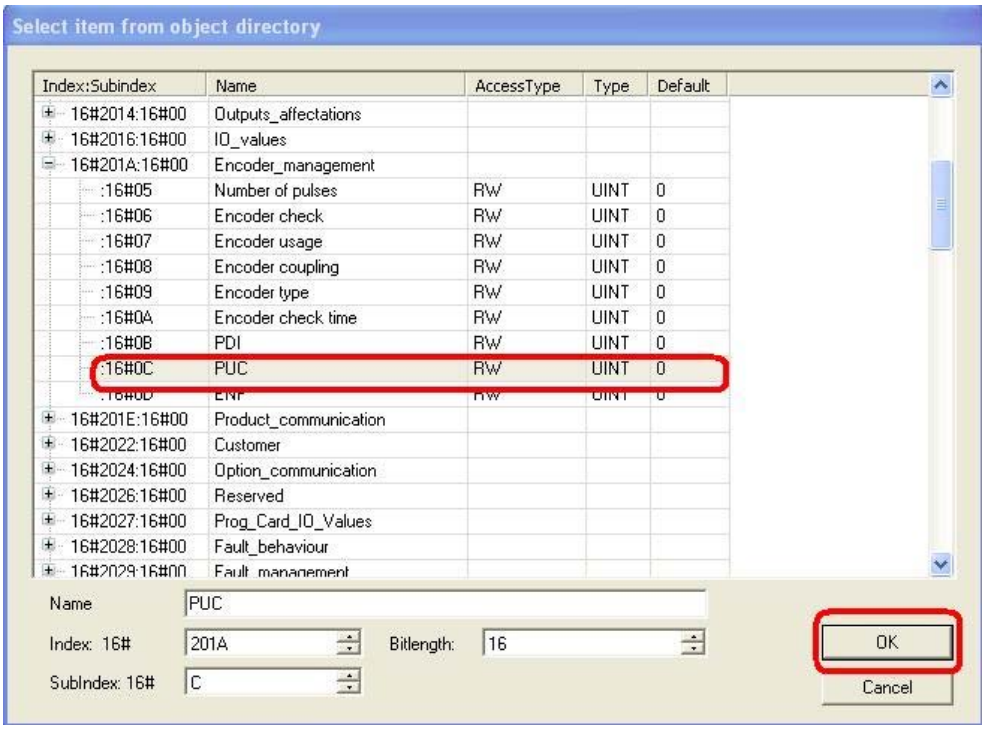
ATV71 CANopen configuration

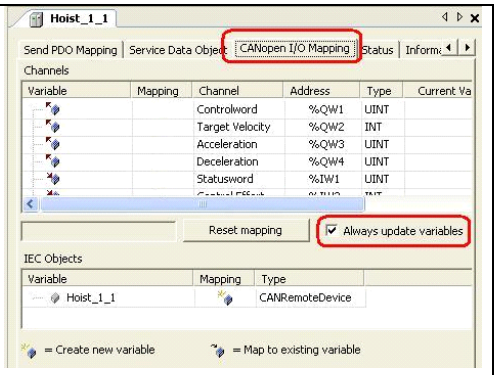
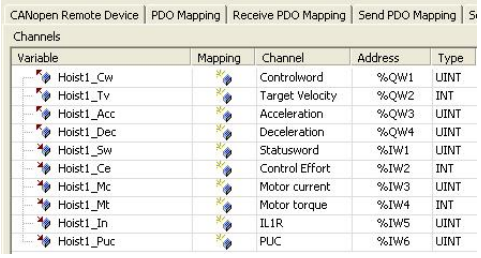
1	The default names of devices are changed in our example. Double click on the **Hoist_1_1**	
2	Set the **Node ID** to **1**	
3	Click on **CANopen Remote Device** tab and check **Enable Expert PDO Settings**	

4	Click on **PDO Mapping** tab 	
5	Check **1st Receive PDO** **1st Transmit PDO** **2nd Transmit PDO** Uncheck the **2nd Receive PDO** 	
6	Click on **Receive PDO Mapping** tab	
7	Expand the **1st Receive PDO**	
8	Click on **1st Receive PDO**	
9	Click on **Add Mapping…**	

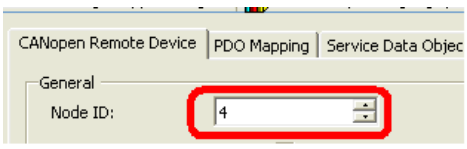
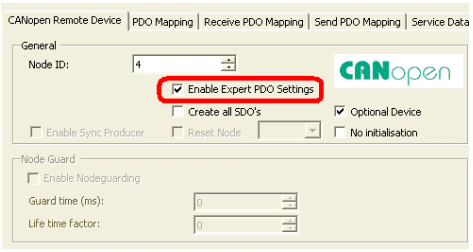
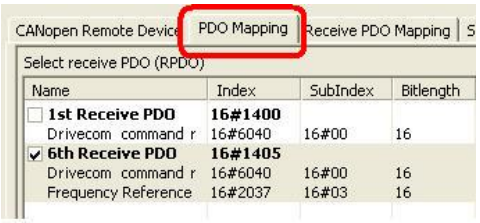
10	Add the parameter **Acceleration** Click on **OK** 
11	Repeat the steps 9 to 10 to add the following parameter **Deceleration**
12	The data available in the **1st Receive PDO** 
13	Click on the **Send PDO Mapping** tab 
14	Expand the **1st Transmit PDO** 
15	Click on **1st Transmit PDO** 
16	Click on **OL1R** 

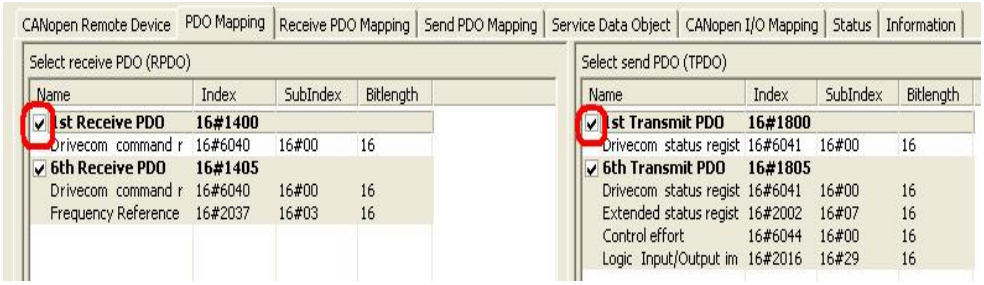
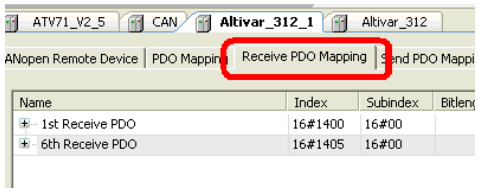
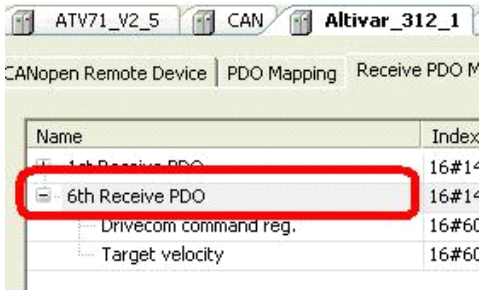
17	**Delete…** this parameter	
18	Click on **Add Mapping…**	
19	Add the parameter **Statusword** and Click on **OK** 
20	Repeat the steps 17 to 18 to add the following parameters **Control Effort** **Motor current** **Motor torque**	
21	The data available in the **1st Transmit PDO**	
22	Expand the **2nd Transmit PDO**	

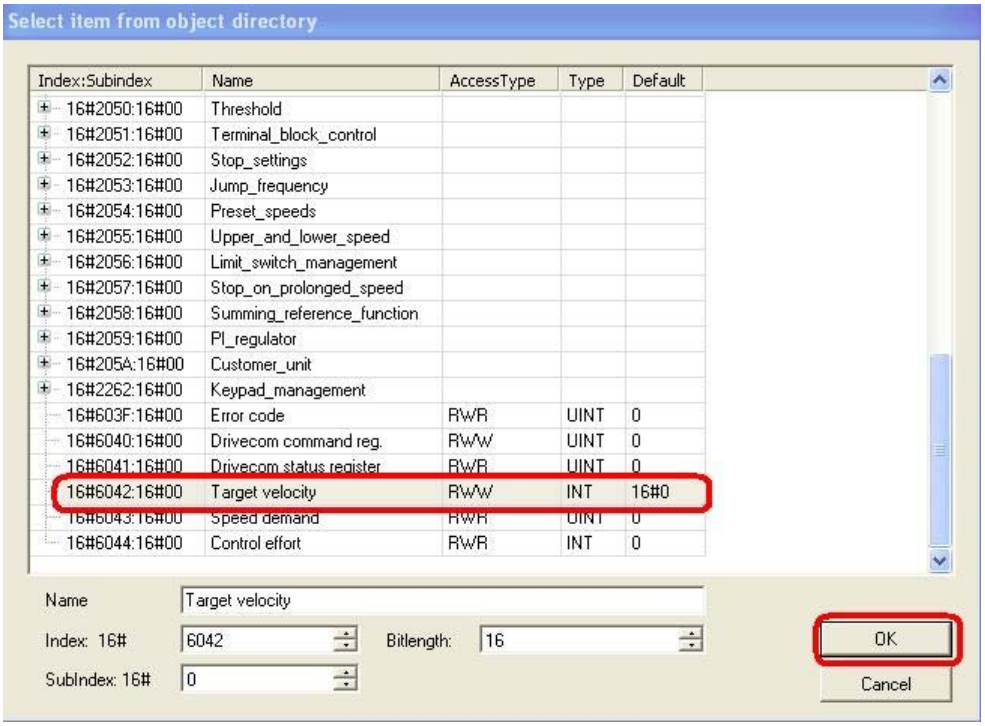
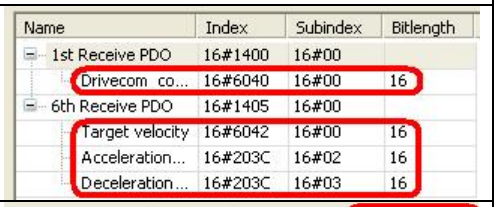
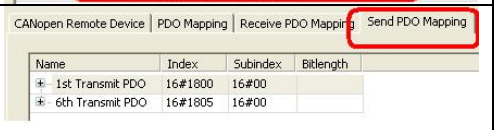
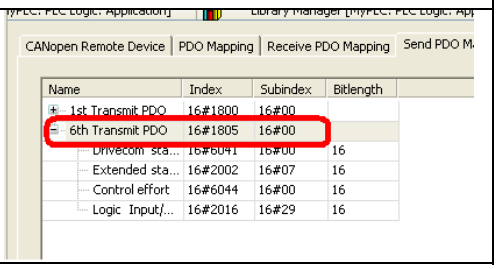
23	Select the patemeter **Statusword** **ETI status word** **Control Effort**	
24	**Delete…** these parameter	
25	Click on **Add Mapping…**	
26	Add the parameter **PUC** (encoder value) Click on **OK**	
27	The data available in the **2nd Transmit PDO**	

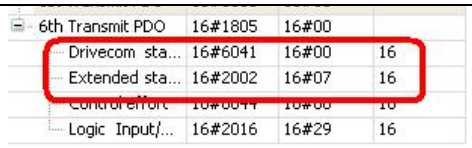
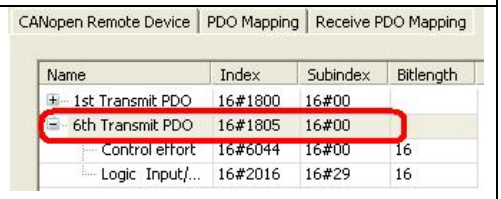
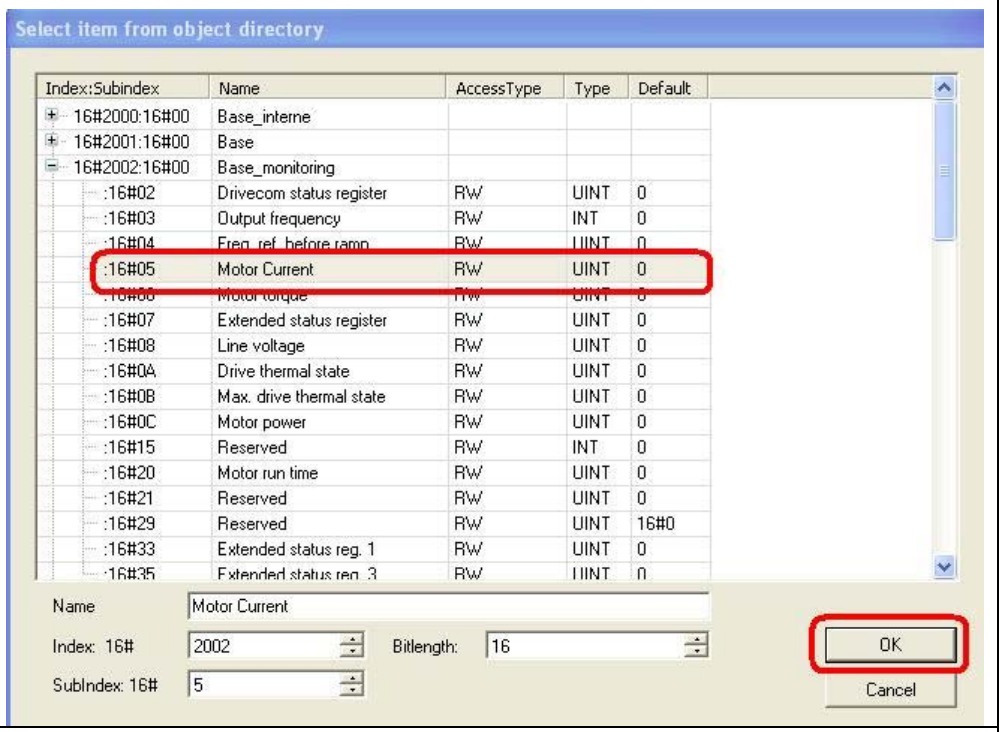
28	Click on **CANopen I/O Mapping** tab Check **Always update variables**	
29	Assign user defined symbolic names in the **Variable** section	
30	**NOTE:** Repeat the same steps with the other ATV71 drive (Hoist_2_2) and change the node ID to **2**	

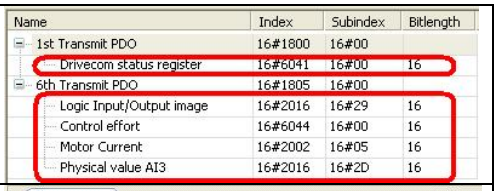
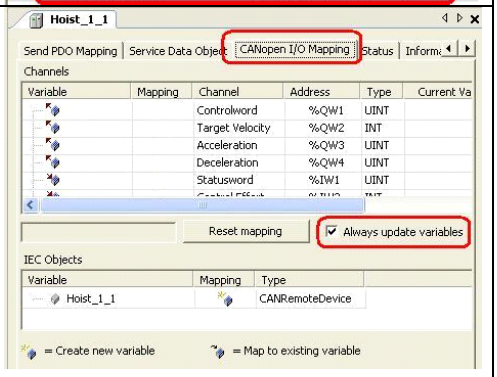
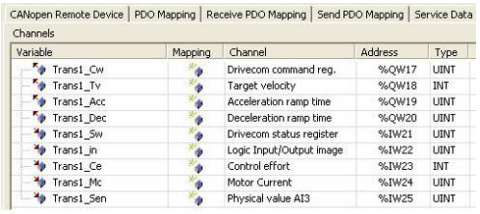
ATV312 CANopen configuration

1	Double click on **Trol_1_4**	
2	Set the **Node ID** to **4**.	
3	Check **Enable Expert PDO Settings**	
4	Click on **PDO Mapping** tab	

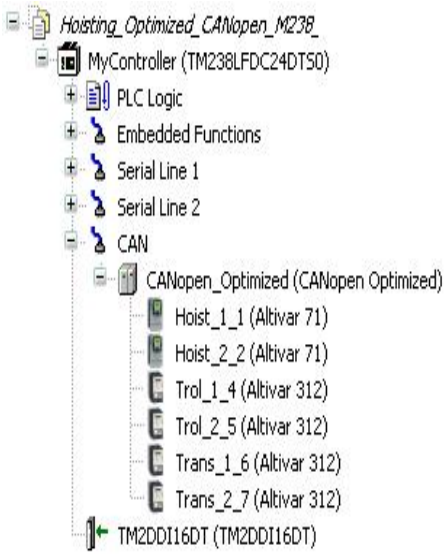
5	Check **1st Receive PDO** **1st Transmit PDO**	
6	Click on **Receive PDO Mapping** tab	
7	Expand the **6th Recieve PDO**	
8	Click on **6th Receive PDO**	
9	Select the parametes **Drivecom command reg.** **Frequency Reference**	
10	Click on **Delete…**	
11	Click on **Add Mapping…**	

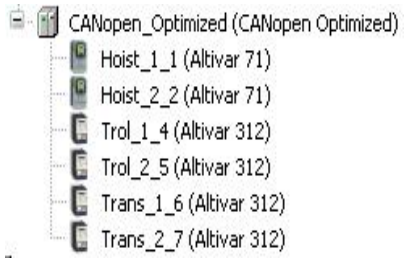
12	Add parameter **Target velocity** Click on **OK** 
13	Repeat the steps 11 to 12 to add the following parameters **Acceleration ramp time** **Deceleration ramp time**
14	The overview of **1st Receive PDO and the** **6th Receive PDO** 
15	Click on **Send PDO Mapping** tab 
16	Expand the **6th transmit PDO** 
17	Click on **6th Transmit PDO** 

18	Select the parameters **Drivecom status register** **Extended status register**	
19	Click on **Delete…**	
20	Click on **6th Transmit PDO**	
21	Click on **Add Mapping…**	
22	Add the parameter **Motor current** and Click on **OK**	
23	Repeat the steps 21 to 22 to add the following parameter **Physical value AI3**	

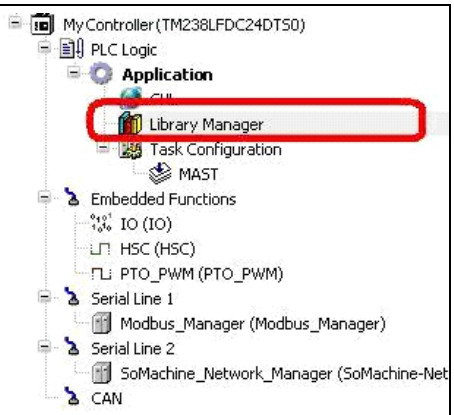
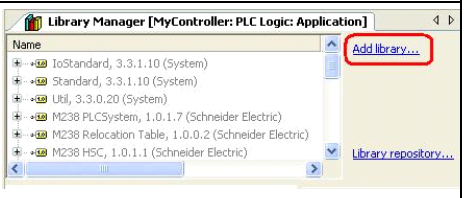
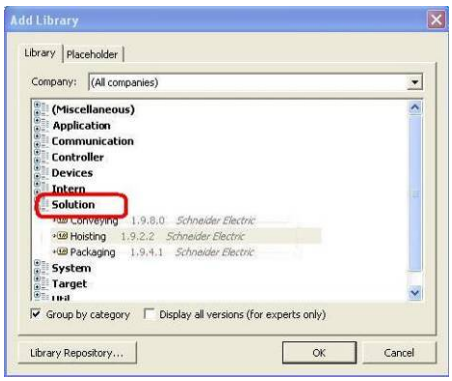
24	The overview of **1st Transmit PDO** and the **6th Transmit PDO**	
25	Click on **CANopen I/O Mapping** tab Check **Always update variables**	
26	Assign user defined symbolic names in the **Variable** section	
27	**NOTE:** Repeat the same steps with the other Altivar 312 drives and change the **Node ID** to **5, 6, 7.**	

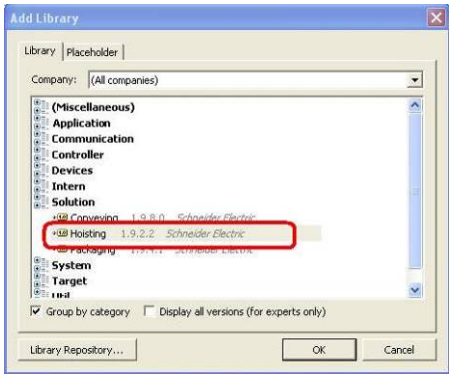
Hardware Layout

1	The picture shows the hardware layout of consists Hardware • **M238 Controller (TM238LFDC24DTS0)** • **2x ATV71 Drives** • **4x ATV312 Drives** • **1x digital TM2 I/O module**	

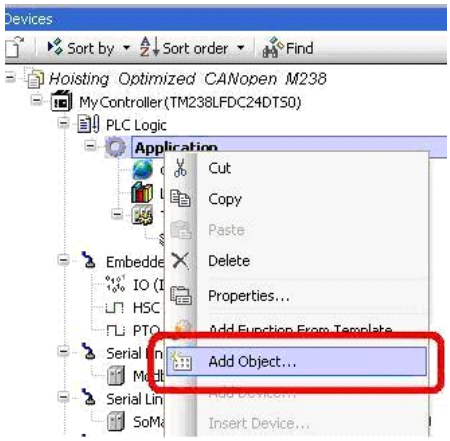
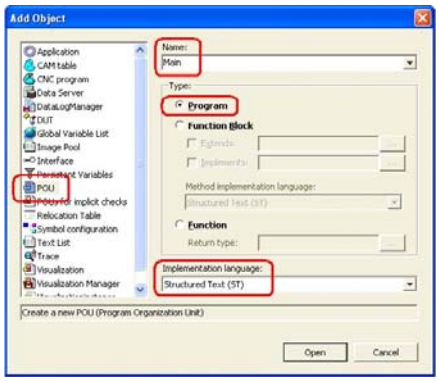
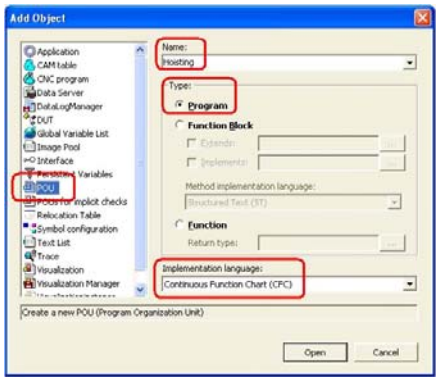
2	In the current project the tag names of the drives changed to • **Hoist_1_1** • **Hoist_2_2** • **Trol_1_4** • **Trol_2_5** • **Trans_1_6** • **Trans_2_7** (the second no. is the Node number)	

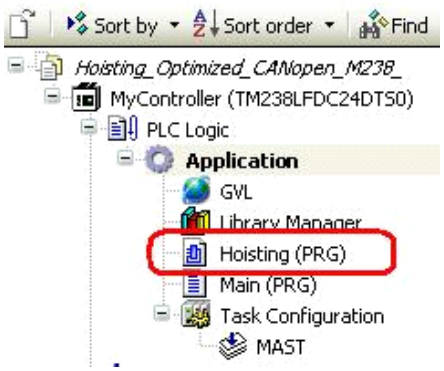
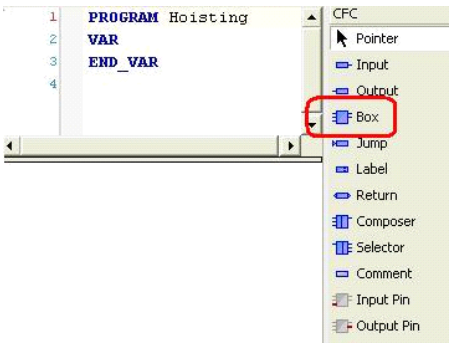
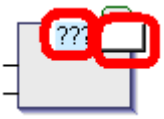
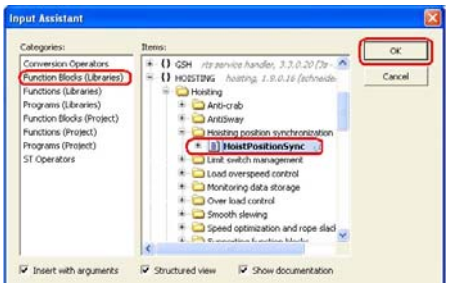
Add Hoisting Library

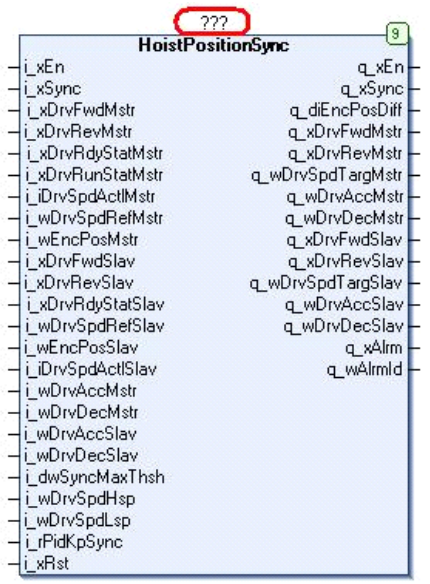
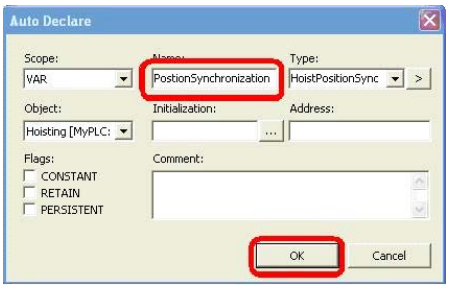
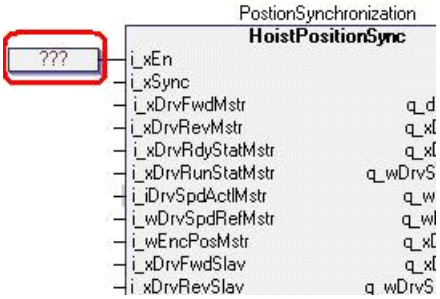
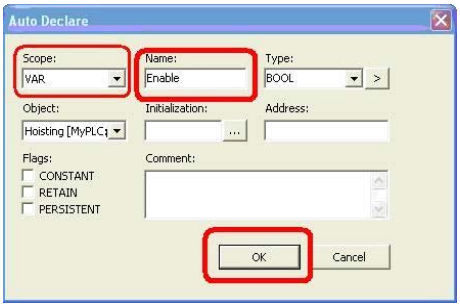
1	Double click on **Library Manager**	
2	Click on **Add library…**	
3	Click on **Solution**	

4	Select **Hoisting** Click on **OK**	

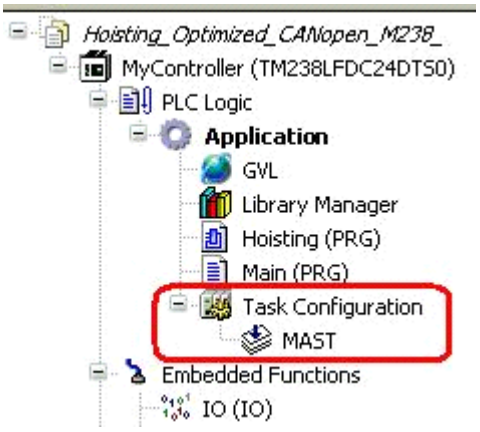
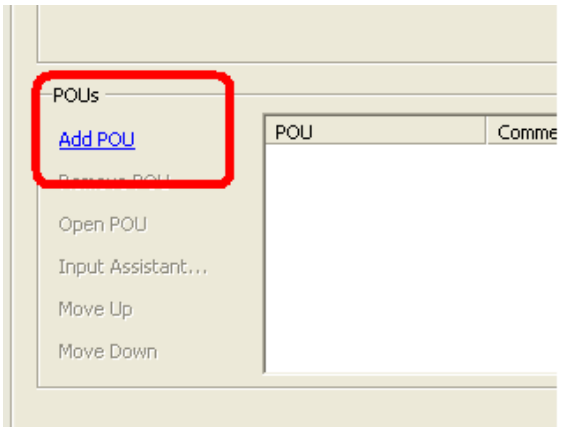
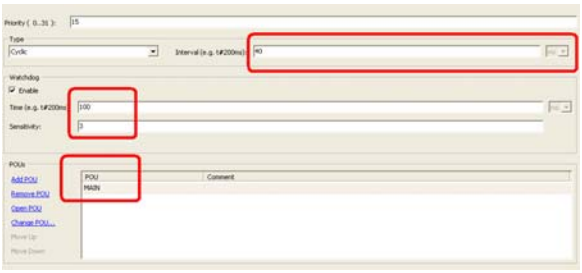
Add POU

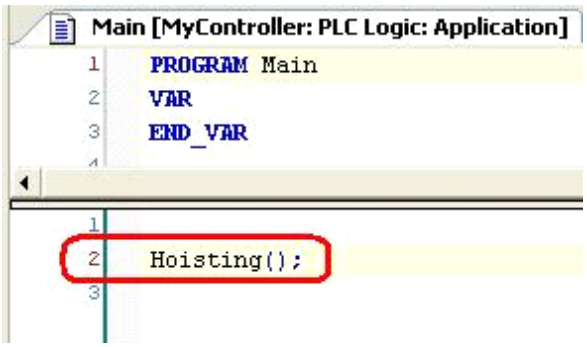
1	Right click on **Application -> Add Object…**	
2	Select **POU** and enter the **Name: Main** **Type: Program** **Implementation language:** **Structured Text (ST)** All the IEC languages can be used for Programs, function blocks and functions. Click on **Open**	
3	Repeat step 1 to add another POU called: **Name: Hoisting** **Type: Program** **Implementation language:** **Continuous Function Chart (CFC)** Click on **Open**	

4	The new **Hoisting** is now visible under **Application**. Double click on **Hoisting** to open it.	
5	The upper frame displays the declaration section. The lower frame is for programming. On the right side is a toolbox. Use drag and drop with the toolbox to place example templates in the programming section, e.g. **Box** for function or function blocks.	
6	Once you have placed a template in the programming section click on the **???**	
7	The will invoke the **Input Assistant** window Select **Function Blocks (Libraries) →** **Hoisting→** **Hoisting position synchronization→** **HoistPositionSync** Click on **OK**	

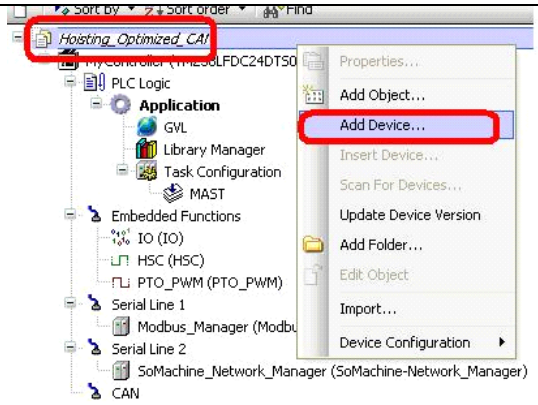
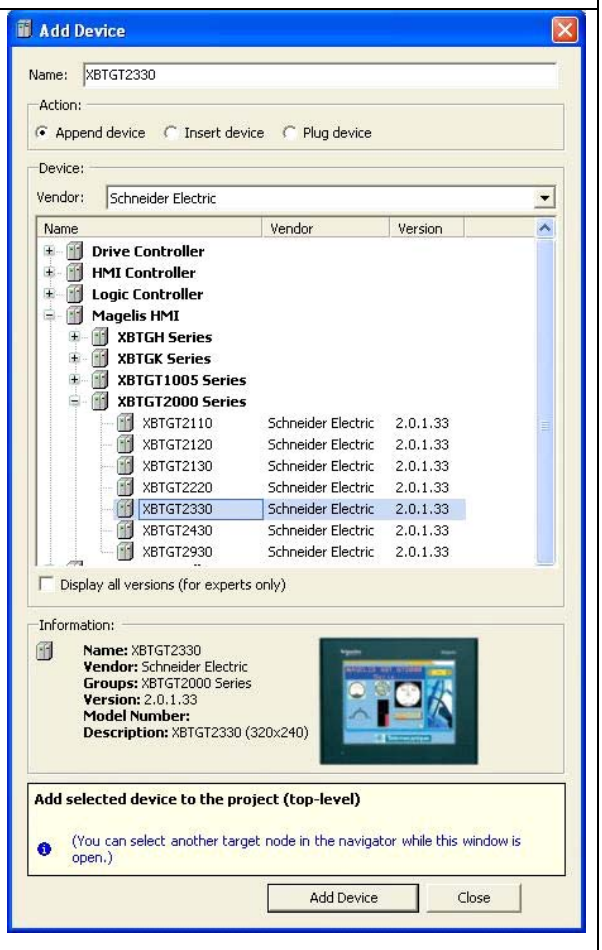
8	The **HoistPositionSync** will appear in the POU. The meaning of the **???** is to show that this function block is still not instantiated. Click on **???**	
9	… and Type a user defined variable name for the Function block. In this case the variable name is **PositionSynchronization** Click on **OK**	
10	Drag and drop from the toolbox an input and place it at the first input of the function block. The connection between both will be drawn with the mouse. Click on **???**	
11	… and create a user defined variable name. **Scope: VAR** (for local variable) **Name: Enable** **Type: BOOL** Click on **OK**	
12	**Complete the application**	

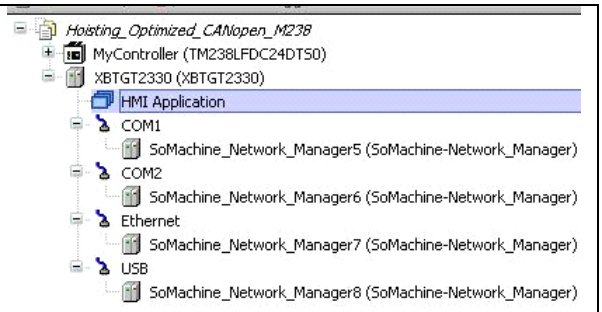
Task Configuration

1	In the **Mast** task of the **Task Configuration** there must be at minimum one POU, otherwise no program code will be invoked cyclically. Double click on **MAST**	
2	Click on **Add POU**	
3	In Input Assistant select **Categories: Programs (Project)** **Items: Main** Click on **OK**	
4	Now the **Main** POU is in the **MAST** task. Change the following parameter by recommended values **Interval: 40** **Time: 100** **Sensitivity: 3**	

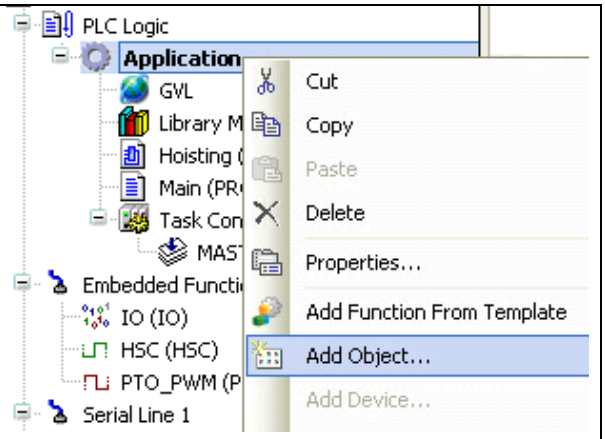
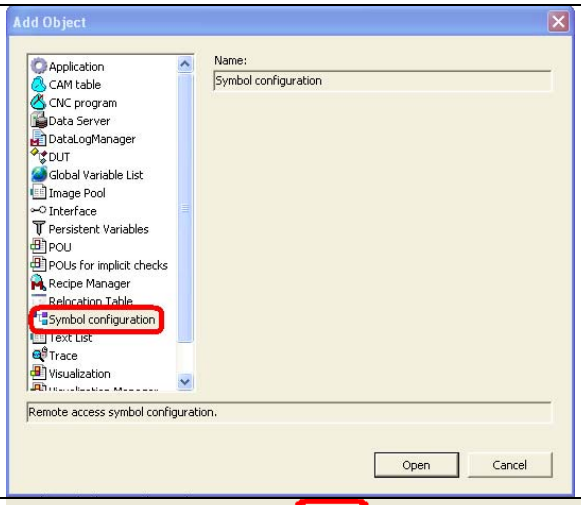
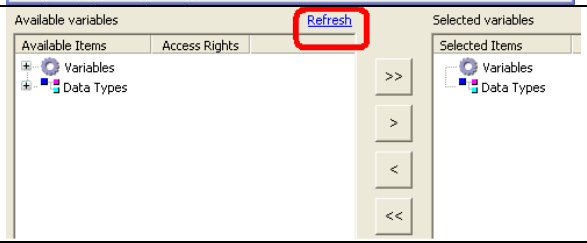
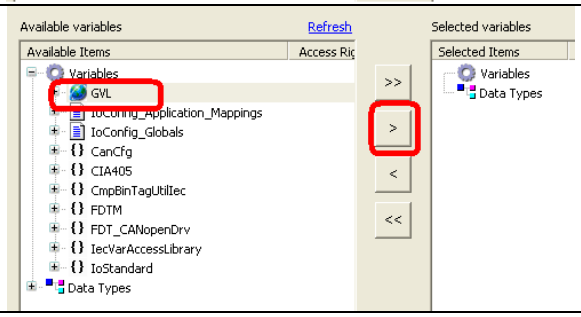
5	The other POU's will be called from the Main POU. Double click on Main (PRG) Enter in body of the POU: **Hoisting( );** Now this program part will be compiled and invoked.	

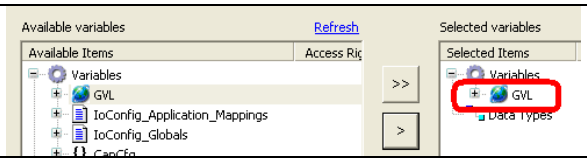
Add Vijeo Designer HMI

1	To add a HMI unit to the project right click on **Hoisting_Optimized_CANopen_M238 →** **Add Device…**	
2	Select **HMI→** **XBTGT 2000 Series→** **XBTGT2330.** Click on **Add Device**	

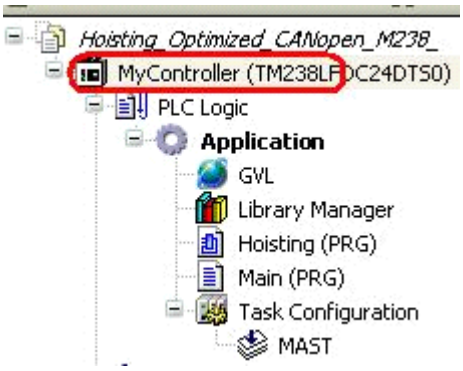
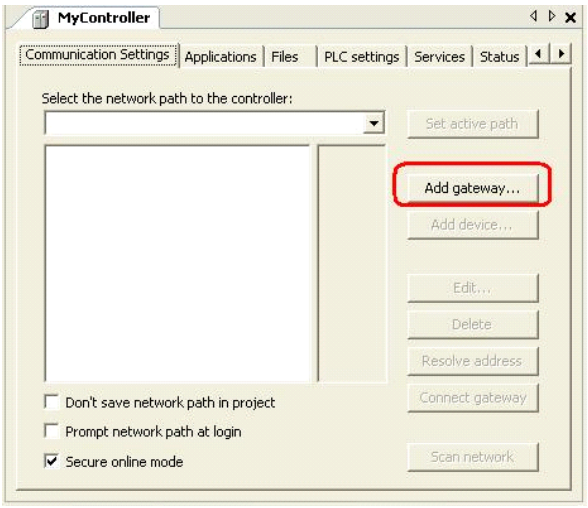
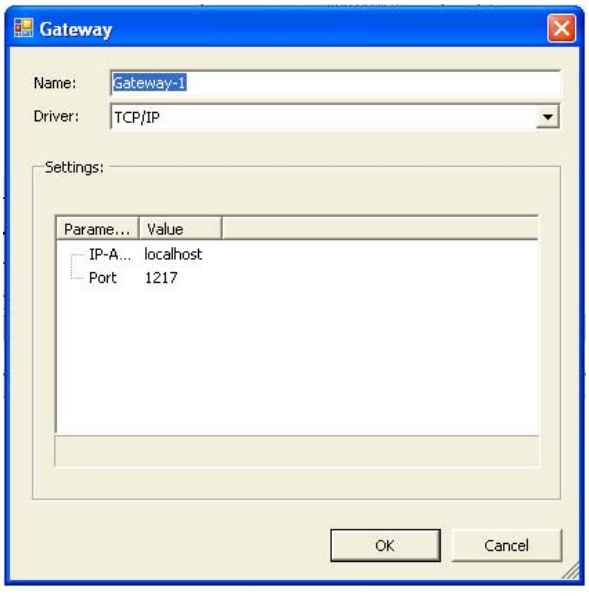
3	The new HMI **XBTGT2330** is now listed in the configuration. With double click on **HMI Application**, the program Vijeo Designer opens and you can start programming. (See chapter HMI)	

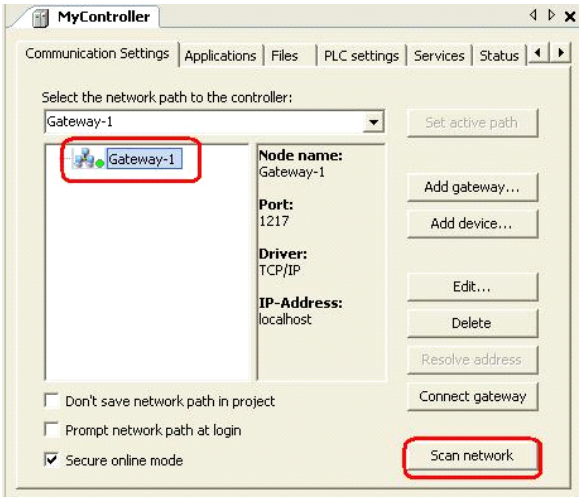
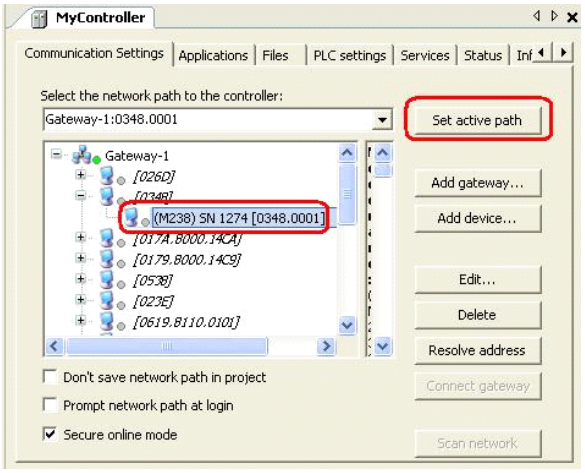
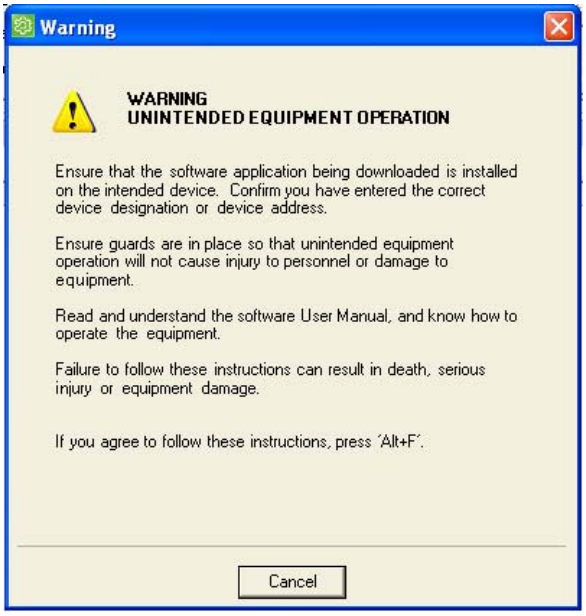
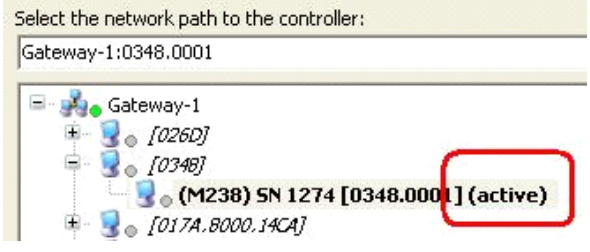
Configure controller ↔ HMI Data Exchange

1	Right click on: **Application→** **Add Object…**	
2	Select **Symbol configuration** Click on **Open**	
3	Click on **Refresh** in the now open Symbol configuration.	
4	All Variables created in the user program are shown in the **Available variables** list. The global variables are located in the **GVL** folder. To export variables to the HMI, select them and click on **>**	

5	The right frame lists the **Selected variables** which are to be used in the HMI.	

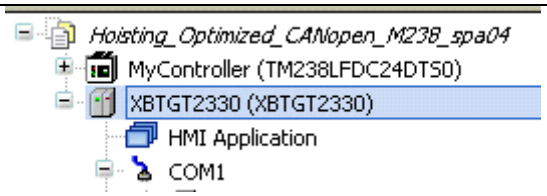
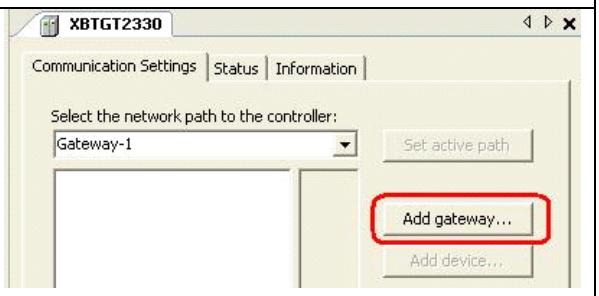
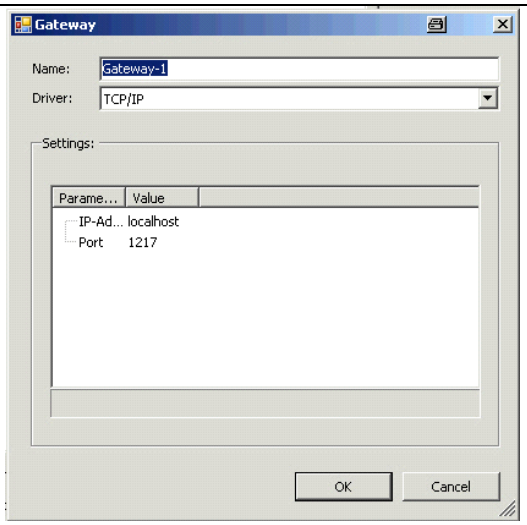
Communication Settings Controller ↔ PC

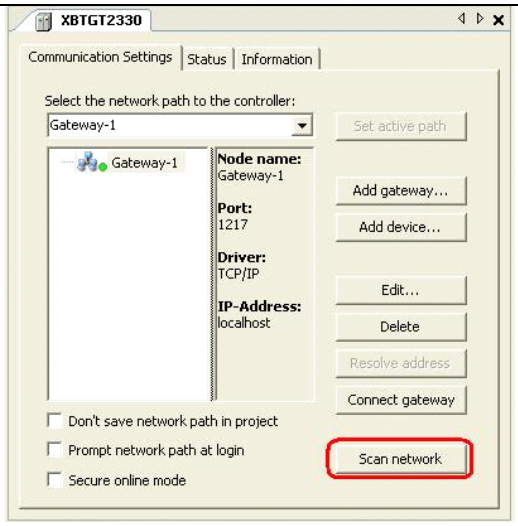
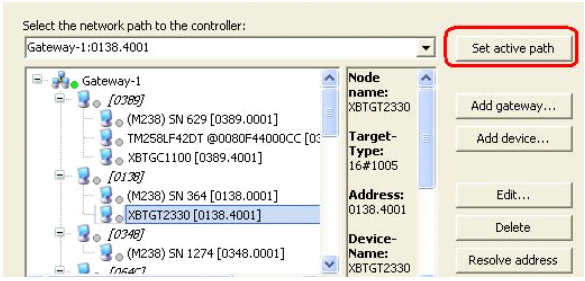
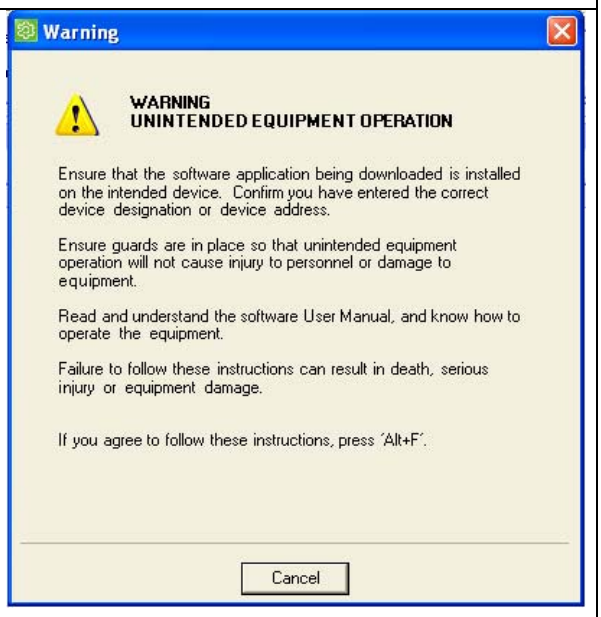
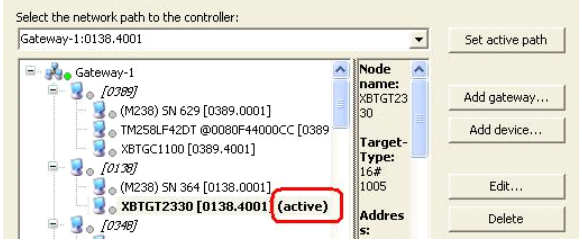
1	Configure the communication gateway Double click on **MyController**.	
2	On **Communication settings** tab click on **Add gateway…**	
3	Keep the factory settings and Click on **OK**	

4	Select **Gateway-1** Click on **Scan network**	
5	During the scan, the **Scan network** button becomes grayed out. When the scan is finished, the **Scan network** button becomes active again and the devices that have been detected are listed under **Gateway-1.** Select the controller that is being used and click on **Set active path.**	
6	A warning pop-up window opens	
7	The used M238 is now marked as **(active)**	

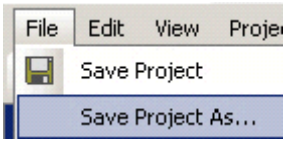
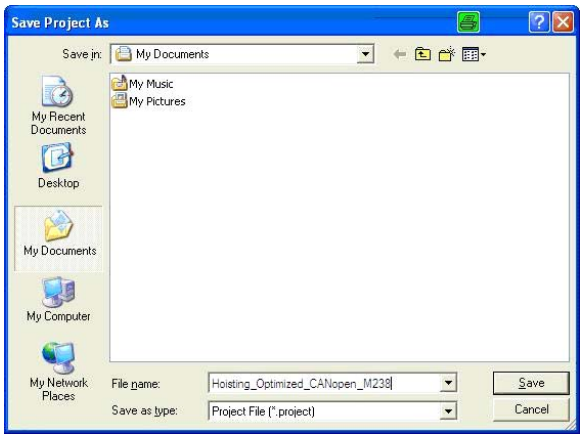
	NOTE: Every M238 has a unique Serial Number that is a part of the default name (in this case: **SN 1274**). If you would like to change the default name of your controller: click on **Edit** In the displayed pop-up window go to the **Device Name** field and enter the new unique name for your controller. In our example we keep the factory setting name.	 

Communication Setting HMI ↔ PC

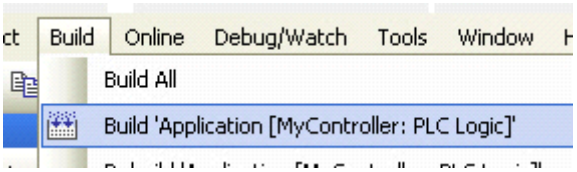
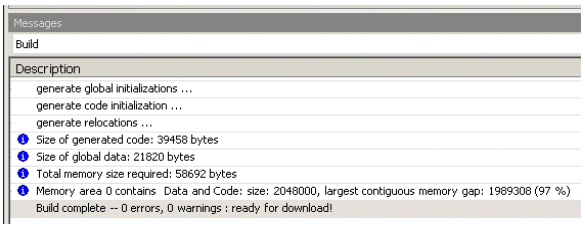
1	To configure the communication gateway Double click on **XBTGT2330**.	
2	On **Communication Settings** tab, Click on **Add gateway...**	
3	Retain the default values and Click **OK.**	

4	Select **Gateway-1** Click on **Scan network**	
5	When the scan is finished, the devices pop up under the gateway. Select the used HMI Click on **Set active path**	
6	A warning pop-up window opens	
7	The used HMI is now marked as **(active)**	

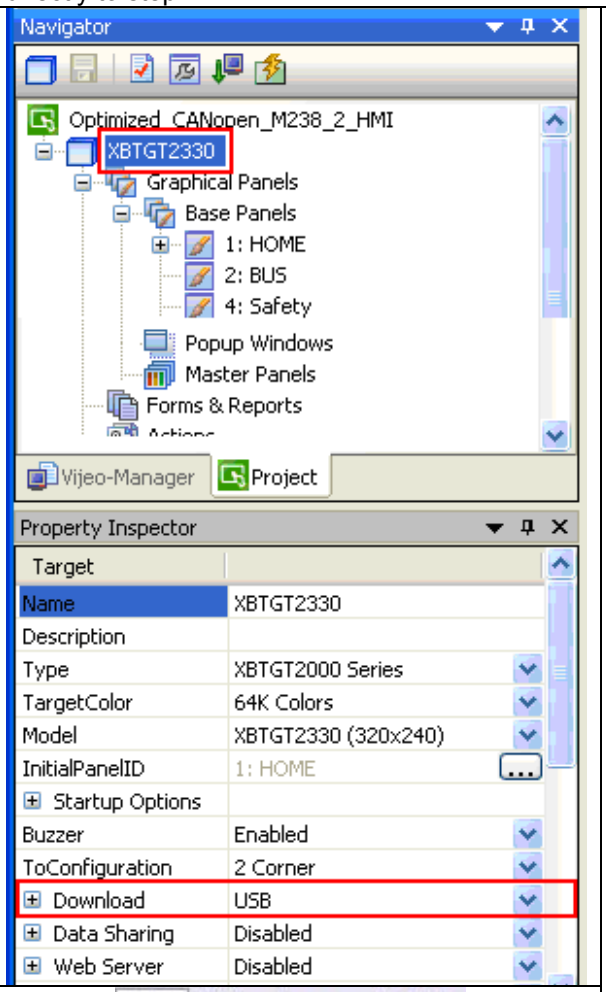
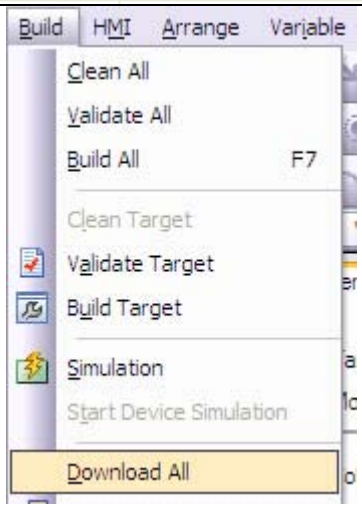
Save the project

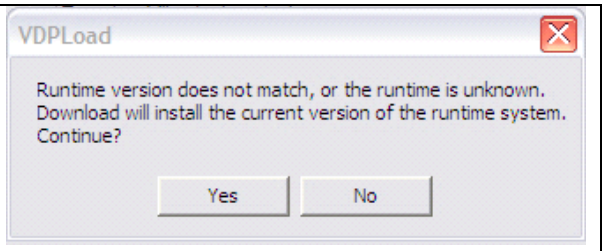
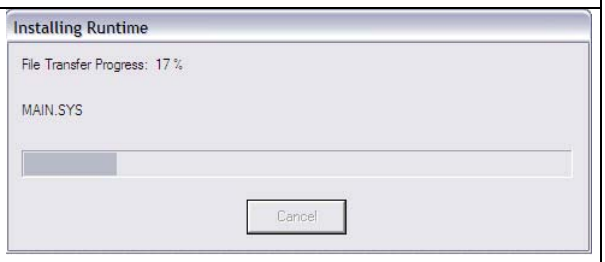
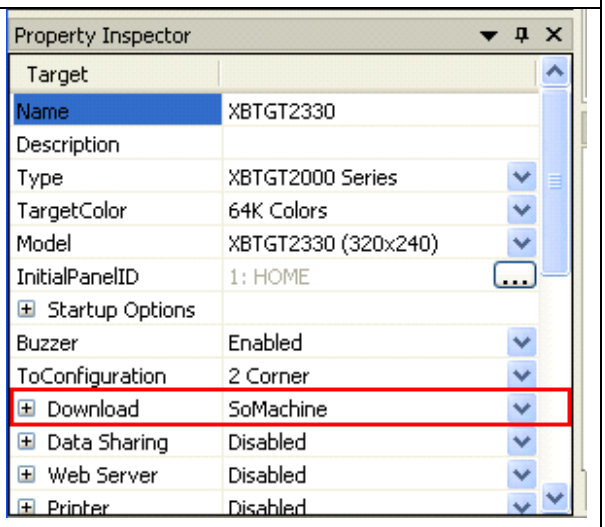
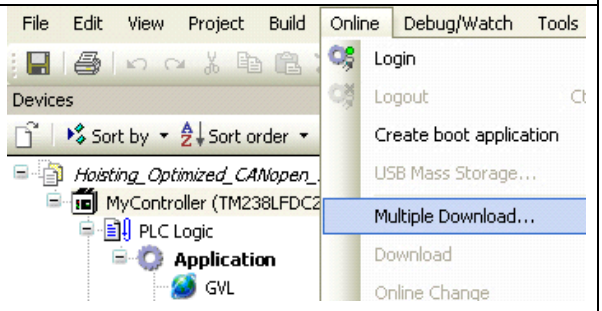
1	To save the project and change the name click **File→** **Save Project As…**	
2	Select the desired location and put a file name. In this case the file name is **Hoisting_Optimized_CANopen_M238** Click on **Save**	

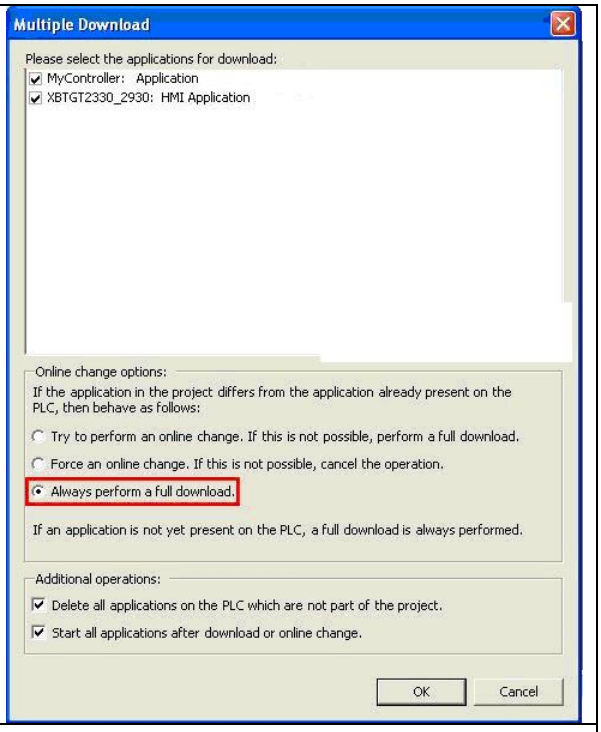
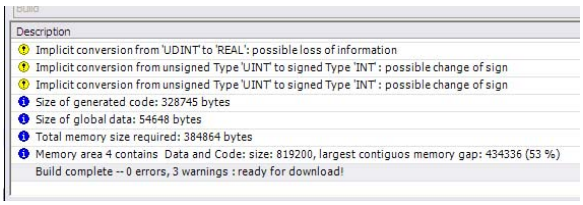
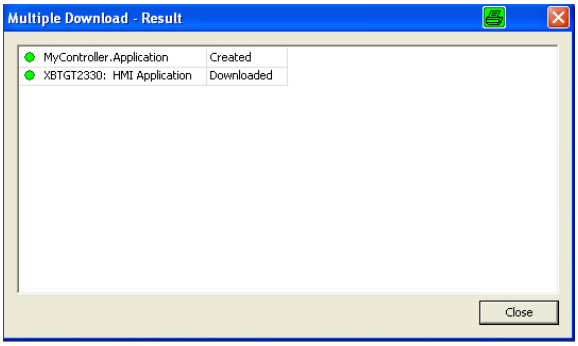
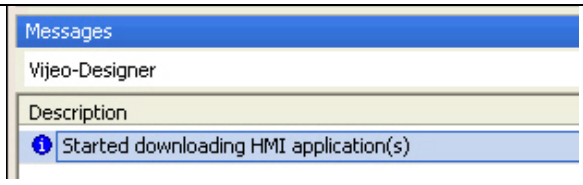
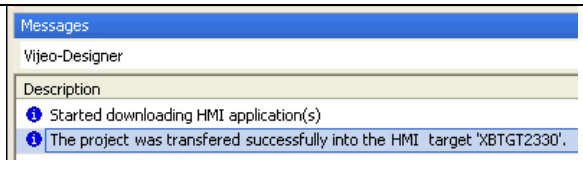
Build Application

1	To build the application click on **Build** **→ Build 'Application** **[MyController:** **PLC Logic]'.** **Note:** If you wish to build the whole project (HMI and Controller) click **Build All**	
2	After the build you are notified in the **Messages** field as to whether the build was successful or not. If the build was not successful there will be a list in the **Messages** field.	

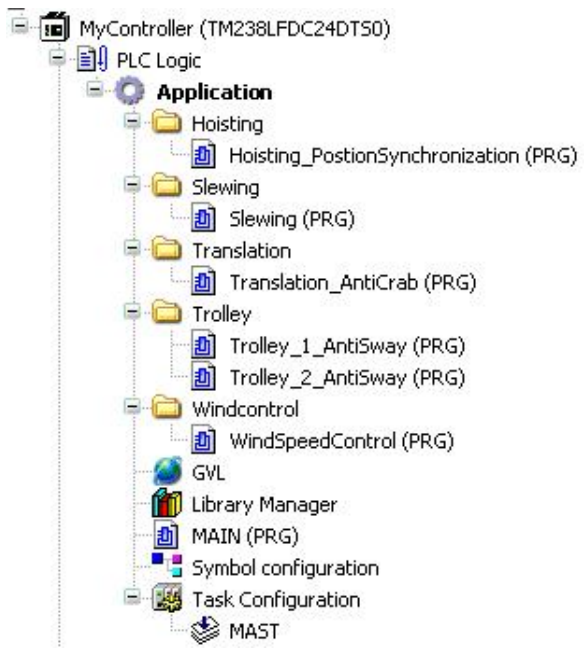
Download the Controller and HMI projects

1	**NOTE:** If it is the first time you are connecting to the HMI you have to first download the latest runtime version to the HMI using Vijeo Designer. This first download is described in the following steps. If this is not the first download go directly to step 7	
2	In Vijeo Designer select the HMI in the device list. Then in the **Property Inspector** select **Download** via **USB**. **Note:** The PC must be connected to the HMI via the cable XBTZG935.	
3	Select: **Build →** **Download All**	

4	The **VDPLoad** dialog indicates that the runtime version does not match. Start the download of the new version Click on **Yes**	
5	The actual state of the download is displayed in a progress bar.	
6	**Once the download is complete, change the** Download **connection in the** Property Inspector **back to** SoMachine.	
7	To download the application to the Controller and the HMI go to SoMachine Program window and select: **Online→** **Multiple Download…**	

8	Check the Controller **MyController**, the HMI **XBTGT2330** and check **Always perform a full download.** click on **OK**.	
9	Before the download starts, a build of the complete project is done. The result of the build is displayed in the Messages window.	
10	The results of the download to the controller are displayed in the Multiple Download – Result window. Here are two examples: In the first dialog, there was downloaded. And in the second dialog, the application was created and downloaded. Click on **Close** to close to the results window.	
11	Once the download to the controller is finished, the HMI download starts.	
12	The result of the HMI download is displayed in the Messages window.	

Application Overview

1	The picture on the right shows the structure of the application. The Application consists • **Hoisting** folder with **Hoisting_PositionSynchronization** as an Application program. • **Slewing** folder with **Slewing** as an Application program. • **Translation** folder with **Translation_AntiCrab** as an Application program. • **Trolley** folder with **Trolley_1_AntiSway** and **Trolley_2_AntiSway** as an Application program. • **Windcontrol** folder with **WindSpeedControl** as an Application program. These entire programs are called in the **MAIN** program.	

HMI

Introduction

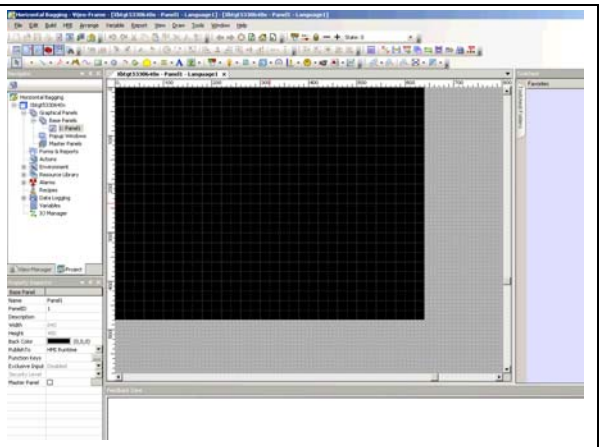
This application uses a Magelis XBTGT2330 HMI. This HMI device communicates via the SoMachine protocol with the M238. The HMI is programmed using the software tool **Vijeo Designer** (delivered with SoMachine), described briefly in the following pages. For the connection between the PC and the HMI Controller use the cable **XBTZG935**.

NOTE:
The Vijeo Designer Tool is opened and closed via SoMachine software. For more information see the chapter
Controller: Add Vijeo Designer HMI

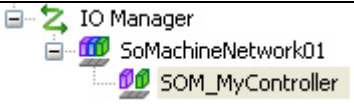
Setting up the HMI is done as follows:

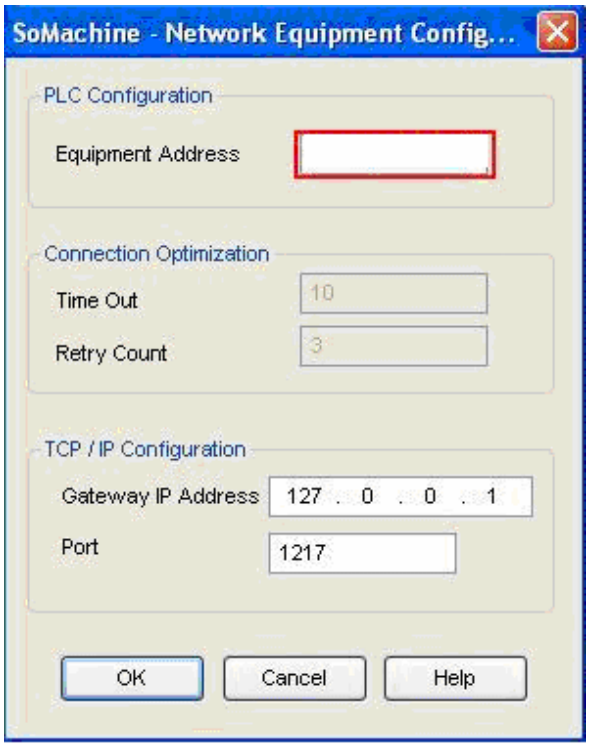
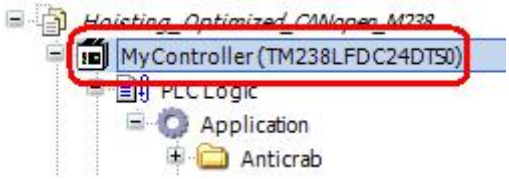
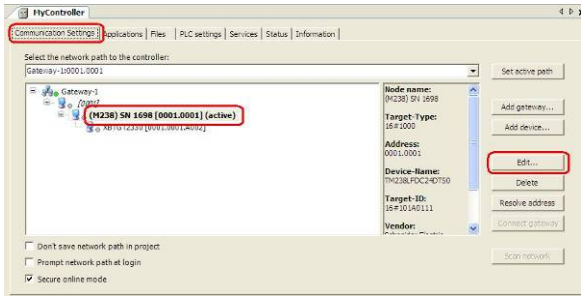
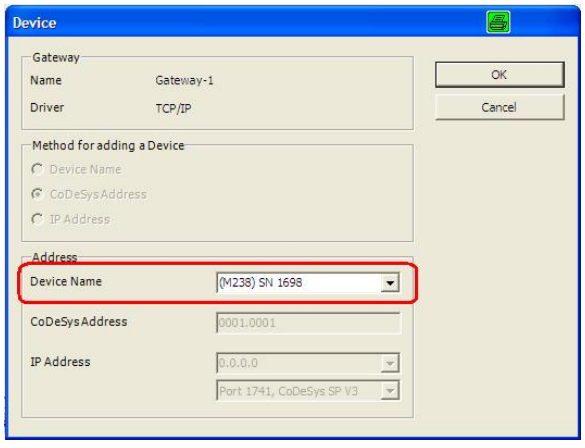
- Main Window
- Communication settings
- Create a switch
- Create a numeric display

Main Window

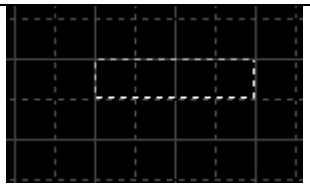
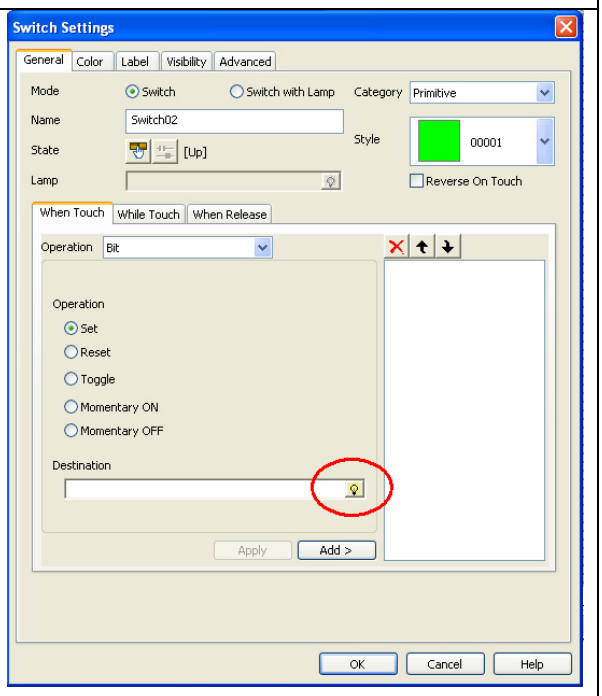
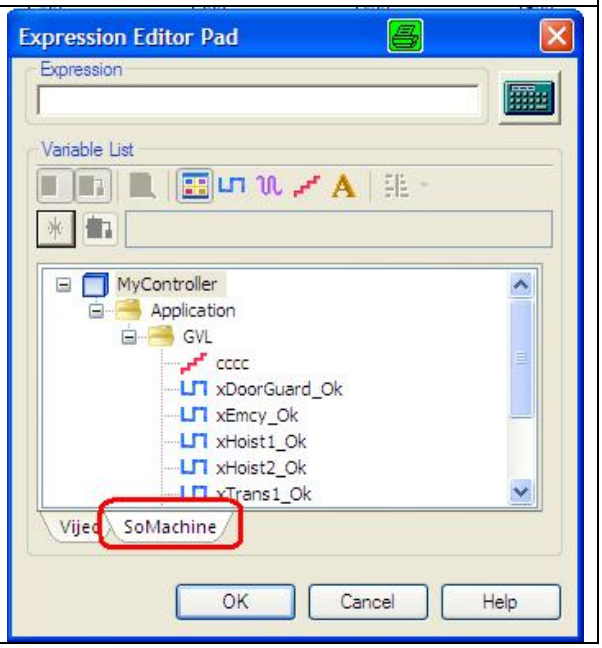
1	After double click on **HMI** application in SoMachine Vijeo Designer creates the HMI program main window.	

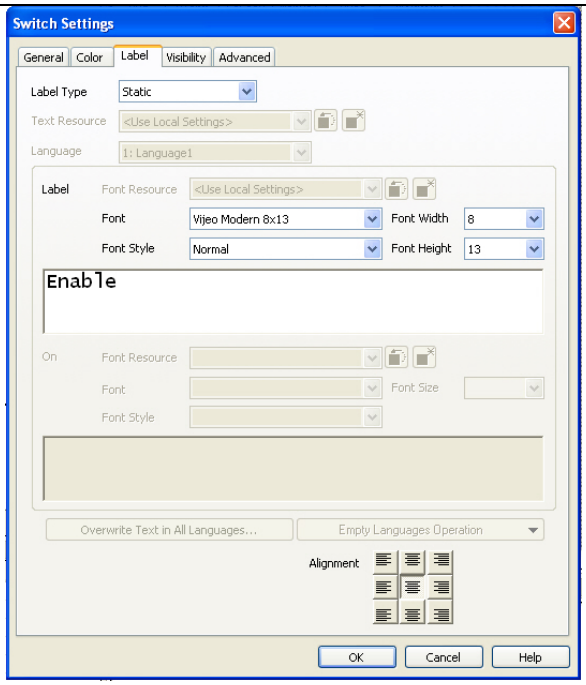
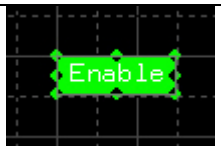
Communi-cation settings

1	To set the communication parameters select in the **Navigator** → **IO Manager** → **SoMachineNetwork01** → double click on **SOM_MyController**	

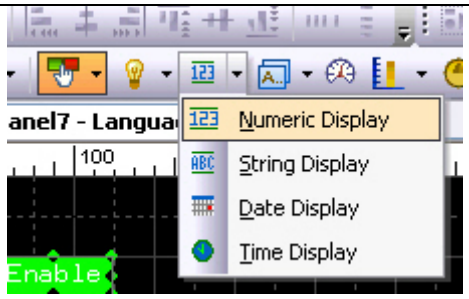
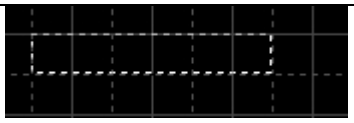
2	In the dialog window, set the controller **Equipment Address.** You will find this address in SoMachine… **(see next step)**	
3	By double clicking the **MyController** in the SoMachine project browser.	
4	In the **Communication** tab select the **M238** and click **Edit…**	
5	The **Equipment Address** of the M238 is displayed under **Device Name**.	

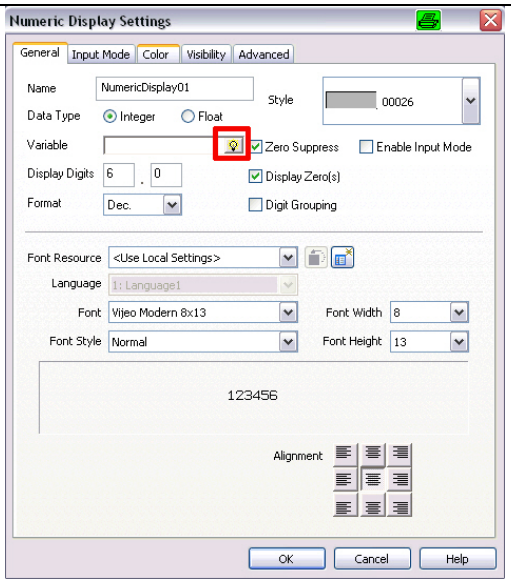
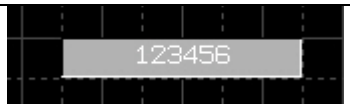
Create a switch

1	Select the **Switch** icon in the **Tool bar**	
2	Select the position and dimension where you wish to place the button by opening a rectangle on the display and pressing enter.	
3	In the **Switch Settings** dialog, select the variable that should be linked (Lamp icon) to the button.	
4	Click on the bulb icon (as indicated in the image above) to open the **Expression Editor Pad** dialog. Use the **SoMachine** tab. Select the required variable and click **OK**.	

5	Go to the **Label** tab. Here select **Label Type: Static** and enter a name for the button, e.g. Enable. Once you have finished your settings click on **OK**.	
6	The display now shows the new button.	

Create a Numeric Display

1	Click on the **Numeric Display** icon in the tool bar.	
2	Select the spot where you want to position the display by opening the rectangle and pressing enter.	

3	In the **Numeric Display Settings** dialog go to the **General** tab. In **Display Digits** you can set the maximum number of the digits to be displayed for both integral and fractional part of the value. To link a **Variable** to the display click on the bulb icon to browse for a variable. Press **OK**.	
4	The display shows the new numeric display.	

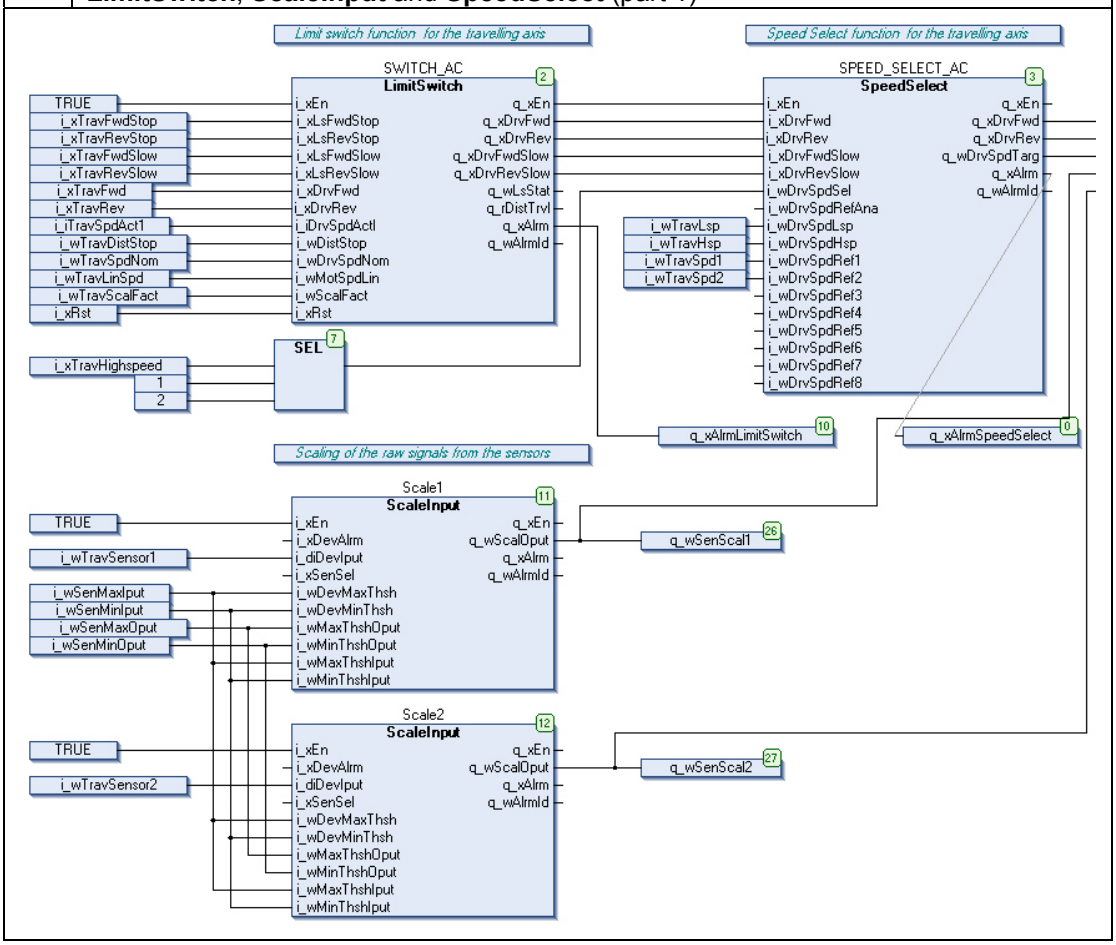
Application functions

Application Functions

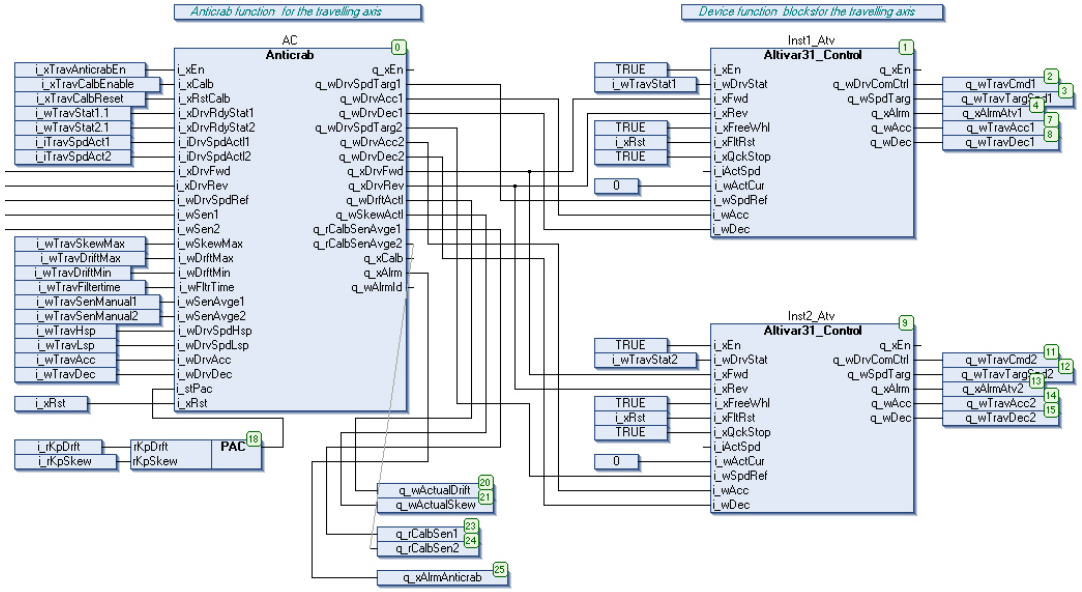
1	The following is a list of application functions describing the operations of the crane movements: • **Anti-crab** within traveling axis • **Anti-sway** within trolley axis • **Load overspeed control** within hoisting axis • **Overload control** within hoisting axis • **Speed optimization and rope slack** within hoisting axis • **Hoisting position synchronization** within hoisting axis • **Monitoring data storage**

Function Anti-crab

1	This screen shot shows the configured function blocks for the traveling axis: **LimitSwitch**, **ScaleInput** and **SpeedSelect** (part 1)

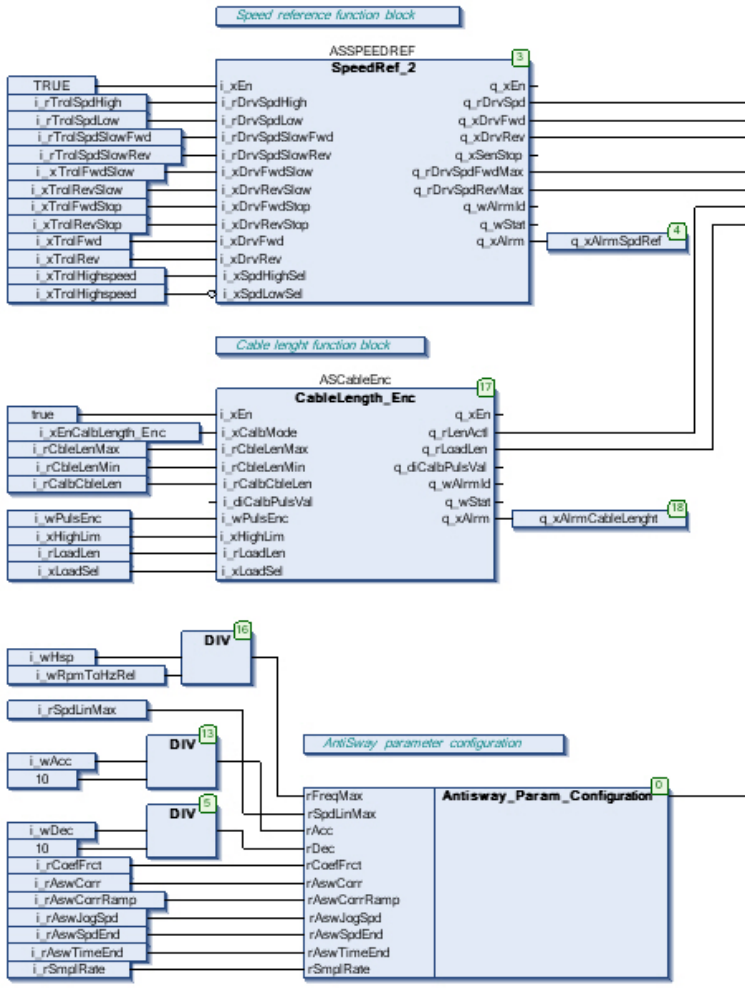


This screen shot shows the configured function blocks for the traveling axis:
Anticrab and **Altivar31_Control** (part 2)

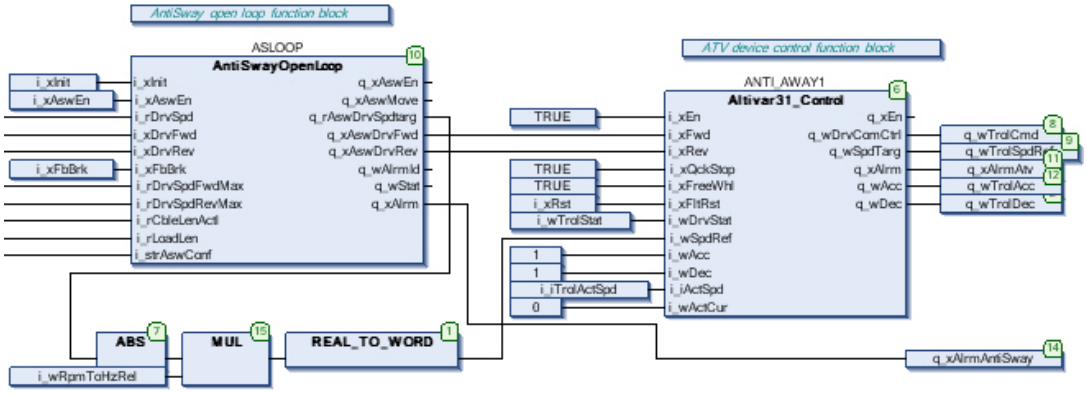


Function Anti-sway

1	This screen shot shows the configured function blocks for the trolley axis: **SpeedRef_2**, **CableLength_Enc** and **Antisway_ Param_Configuration** (part 1)

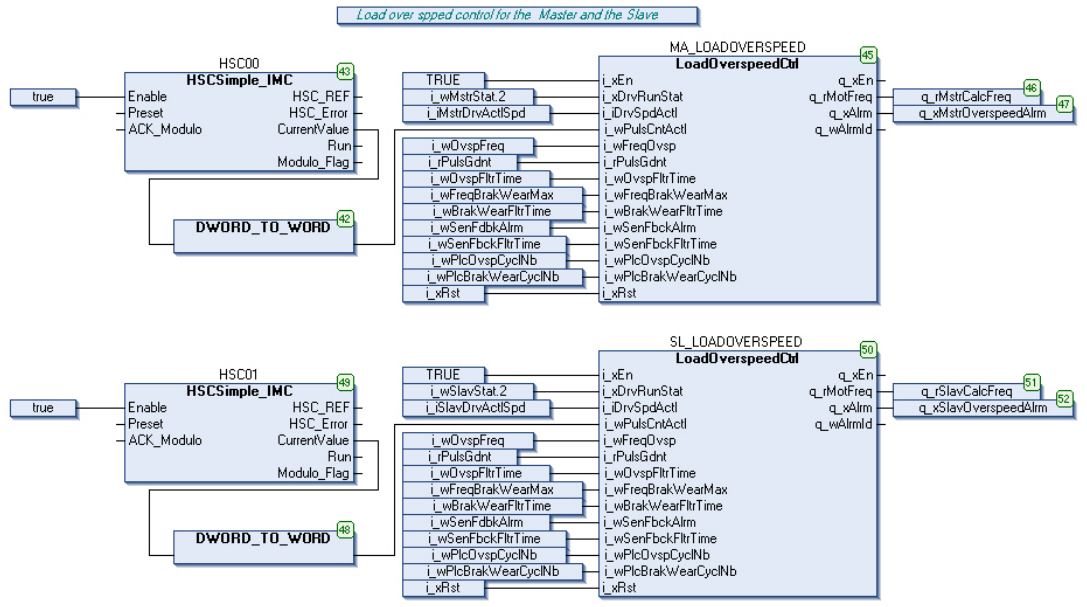


This screen shot shows the configured function blocks for the trolley axis:
AntiSwayOpenLoop and **Altivar31_Control** (part 2)



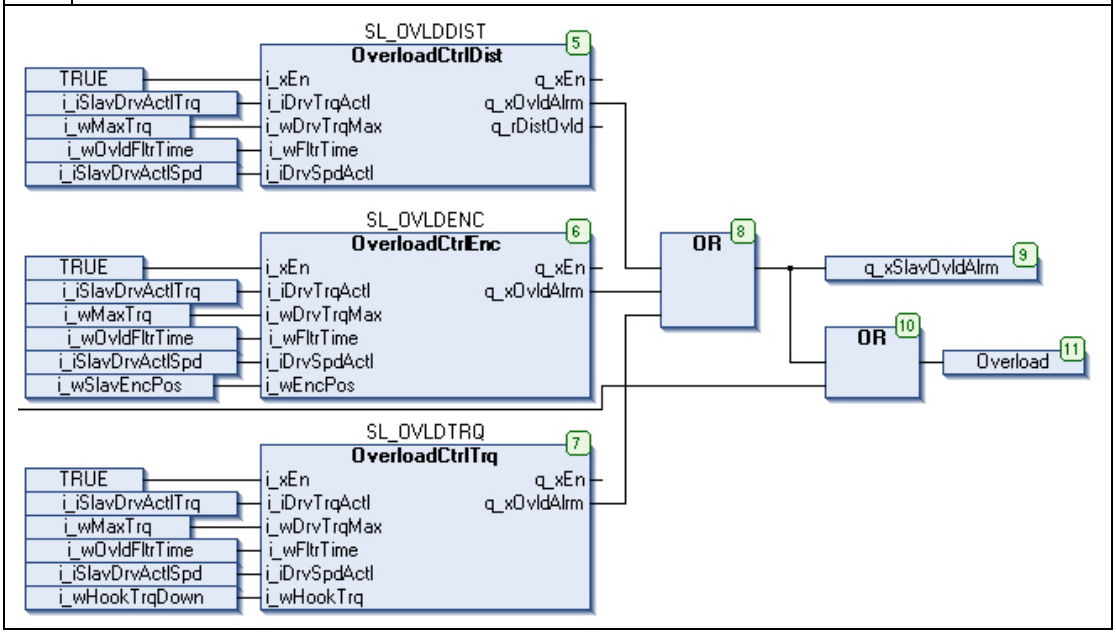
Function Load overspeed control

1	This screen shot shows the configured function blocks for the hoisting axis: **LoadOverspeedControl**



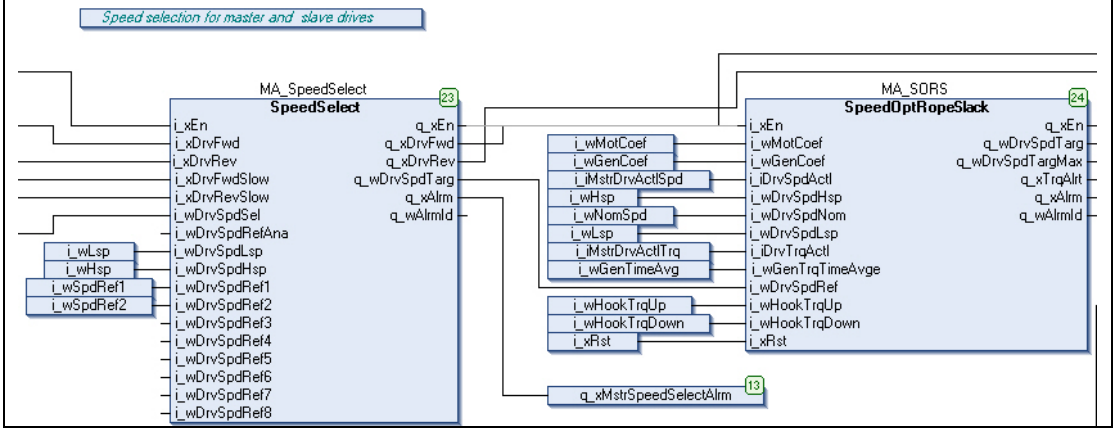
Function Overload control

1	This screen shot shows the configured function blocks for the hoisting axis: **OverloadCtrlDist, OverloadCtrlEnc** and **OverloadCtrlTrq**



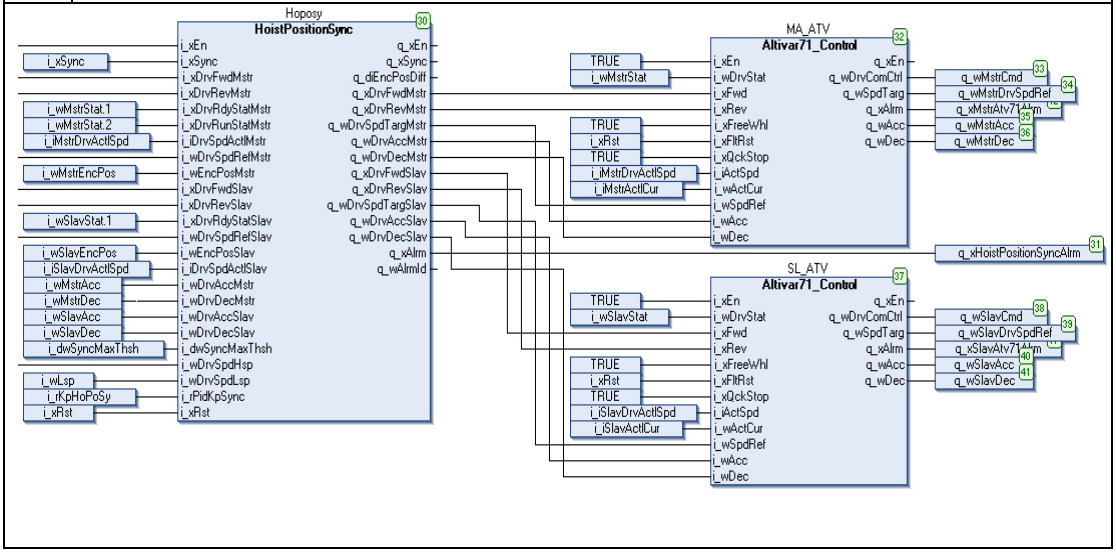
Function Speed optimization and rope slack

1	This screen shot shows the configured function blocks for the hoisting axis: **SpeedSelect** and **SpeedOptRopeSlack**



Function Hoisting position synchronization

1	This screen shot shows the configured function blocks for the hoisting axis: **HoistPositionSync** and **Altivar71_Control** Note : **Altivar71_Control** is part of Altivar library



Devices

Introduction

This chapter describes the steps required to initialize and configure the different devices required to attain the described system function.

General

Altivar 312 and Altivar 71 drives are configured by using the local control panel.

Note

If this is not a new drive it is recommended to return to the factory settings. If you need instructions on how to do this, please read the drive documentation.

It is recommended that the controller is in stop mode before parameterizing the drives.

Altivar 312

Introduction

The ATV312 parameters can be entered or modified via the local control panel on the front of the device.

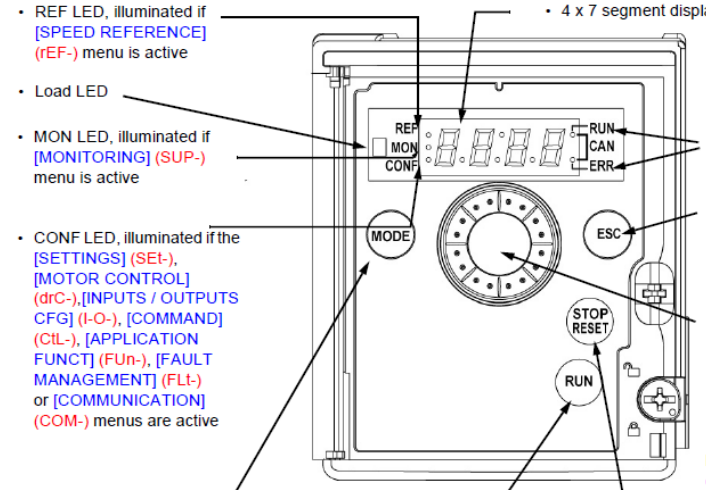
Note

If this is not a new drive it is recommended to return to the factory settings. If you need instructions on how to do this, please read the drive documentation.

Jog dial that is a part of the local control panel and can be used for navigation by turning it clockwise or counter-clockwise. Pressing the jog dial enables the user to make a selection or confirm information.

Control panel

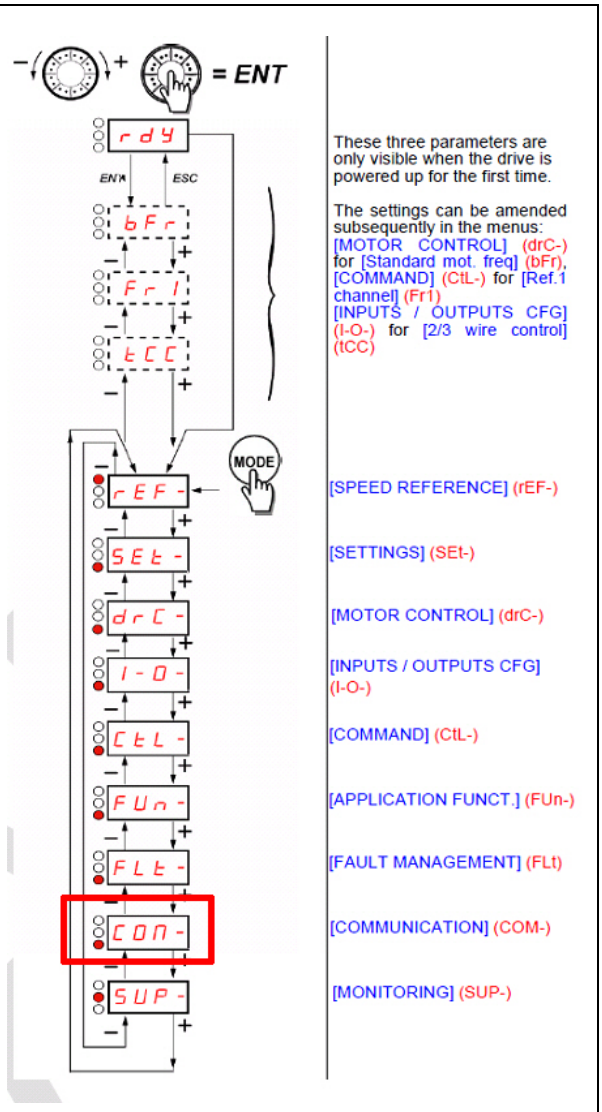
1	The drive parameters can be input using the buttons and the jog dial on the control panel of the Altivar.



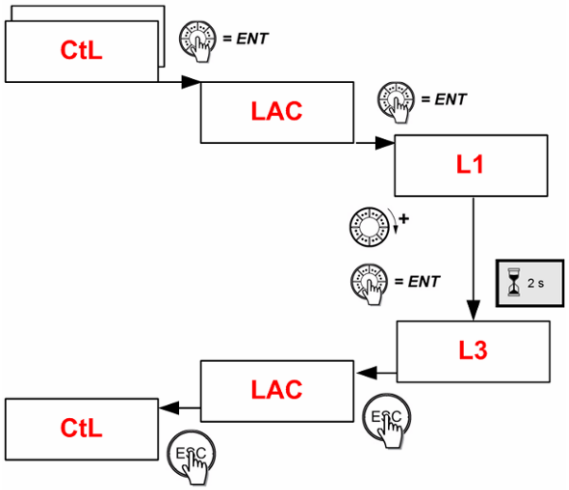
- REF LED, illuminated if [SPEED REFERENCE] (rEF-) menu is active
- Load LED
- MON LED, illuminated if [MONITORING] (SUP-) menu is active
- CONF LED, illuminated if the [SETTINGS] (SEt-), [MOTOR CONTROL] (drC-),[INPUTS / OUTPUTS CFG] (I-O-), [COMMAND] (CtL-), [APPLICATION FUNCT] (FUn-), [FAULT MANAGEMENT] (FLt-) or [COMMUNICATION] (COM-) menus are active
- MODE button (1): If [SPEED REFERENCE] (rEF-) is displayed, this will take you to the [SETTINGS] (SEt-) menu. If not, it will take you to the [SPEED REFERENCE] (rEF-) menu.
- 4 x 7 segment display
- RUN button: Controls powering up of the motor for forward running if the [2/3 wire control] (tCC) parameter in the [INPUTS / OUTPUTS CFG] (I-O-) menu is set to [Local] (LOC), page 45
- 2 CANopen status LEDs
- Used to quit a menu or parameter or to clear the value displayed in order to revert to the value in the memory
- Jog dial - can be used for navigation by turning it clockwise or counter-clockwise - pressing the jog dial enables the user to make a selection or confirm information. = *ENT*

  Functions as a potentiometer if [Ref.1 channel] (Fr1-) in the [COMMAND] (CtL-) menu is set to [Image input AIV1] (AIV1)
- STOP/RESET button
  - Enables detected fault to be reset
  - Can be used to control motor stopping
    - If [2/3 wire control] (tCC) is not set to [Local] (LOC), freewheel stop
    - If [2/3 wire control] (tCC) is set to [Local] (LOC), stop on ramp or freewheel stop during DC injection braking

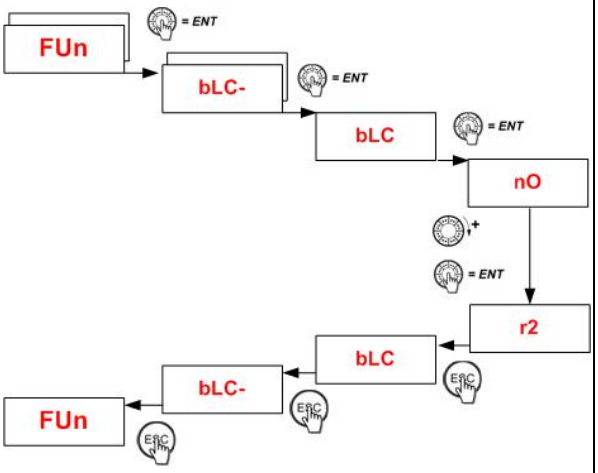
CANopen settings

1	Using the buttons on the front panel, select the sub-menu **Communication.**	
2	In the **Communication (COM)** sub-menu input the CANopen address in the parameter **AdC0.** In the example application the addresses for the four drives are **4** to **7**	
3	Also in the **Communication (COM-)** sub-menu, in the parameter **BdC0,** set the Baudrate to **500.0** (kBits).	
4	For the ATV312 to operate with the new address or Baudrate, a power cycle (on, off, on) is required.	

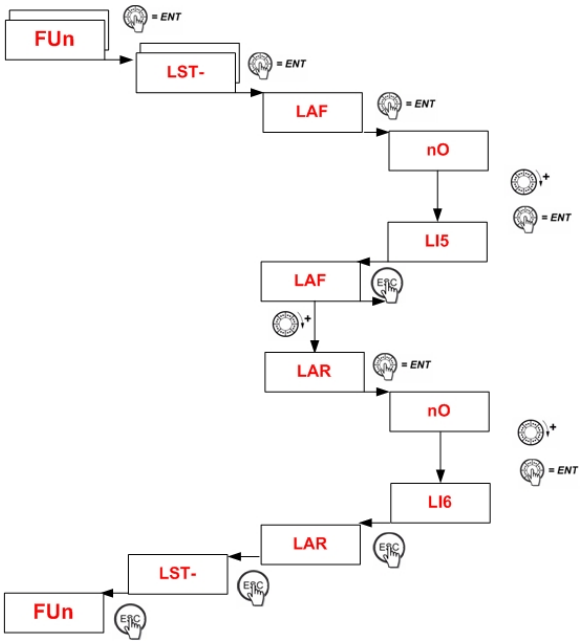
Changing the Access Level LAC

1	To set the parameters for the brake function a higher access level (L3) is required.	
2	To go to expert mode L3: ➔ Select **CtL-** [COMMAND] and press enter ➔ Select **LAC** [ACCESS LEVEL] and press enter ➔ **L1** (Level 1) is displayed ➔ Select **L3** (Level 3) and press enter for 2 seconds to set the new level. Return to the **LAC** with **ESC.** Return to the **CtL** with **ESC.**	

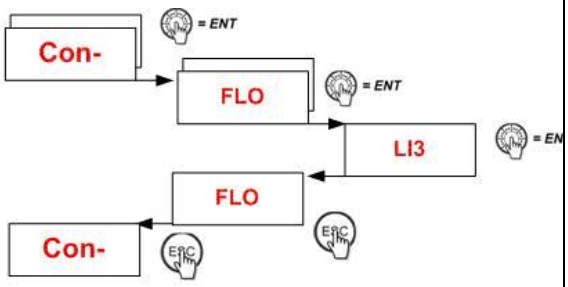
Brake settings

1	The r2 relay output is used for brake control.	
2	To assign the r2 relay output : ➔ Select **FUn-** [APPLICATION FUNCT.] and press enter ➔ Select **bLC-** [BRAKE LOGIC CONTROL] and press enter ➔ Select **bLC** [BRAKE LOGIC CONTROL] and press enter ➔ Select **r2** and press enter. Set the parameters to the values shown here on the right. **Note:** These parameters are for the test machine only. They are **NOT VALID** for every machine. After all parameters are set return to the **bLC** with **ESC.** Return to the **bLC-** with **ESC.** Return to the **FUn** with **ESC.**	 **Brake Logic parameters** **Parameter** / **Value** bLC [Brake assignment] — R2 brL [Brake release freq] — 0 Hz Ibr [Brake release I FW] — 0 A brt [Brake release time] — 0 s LSP [Low speed] — 0 Hz bEn [Brake engage freq] — 0 Hz bEt [Brake engage time] — 0.5 s bIP [Brake impulse] — No

Limit switches configuration

1	The input LI5 and LI6 are assigned to Limit switches	
2	To assign the inputs LI5 and LI6 for the forward and reverse limit switches : ➔ Select **FUn-** [APPLICATION FUNCT.] and press enter ➔ Select **LST-** [Limit switches.] and press enter ➔ Select **LAF** [Forward limit switch .] and press enter. ➔ Select **LI5** and press enter. ➔ Press **ESC** ➔ Select **LAr** [Reverse limit switch .] and press enter. ➔ Select **LI6** and press enter. Once the the limitswitches are configured return to the **LST-** with **ESC.** Return to the **FUn-** with **ESC.**	

Local mode configuration

1	The input LI3 is assigned to Local mode	
2	To assign LI3 for the local mode configuration ➔ Select **Con-** [Communication] and press enter ➔ Select **FLO** [Forced local mode.] and press enter ➔ Select LI3 and press enter. Once the the local mode is configured return to the **FLO** with **ESC.** Return to the **Con-** with **ESC.**	
	Local mode: Local mode is required to manage the axis movement when the wiring is directly connected to the Altivar and not to the controller. Local mode is used to test the axis during: • Commissioning of the crane for the first time; in this case the operator is able to check that the movement is correct. • Maintenance, when it is required to move an axis without the help of the controller.	

Altivar 71

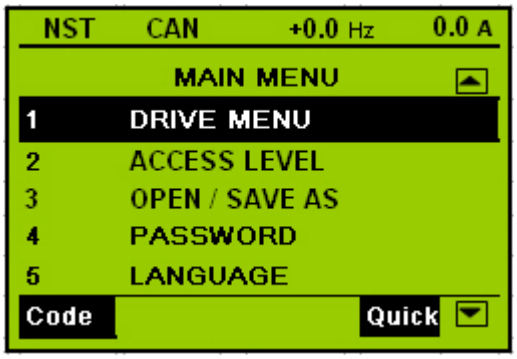
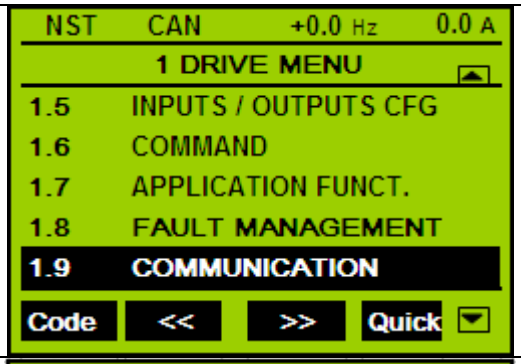
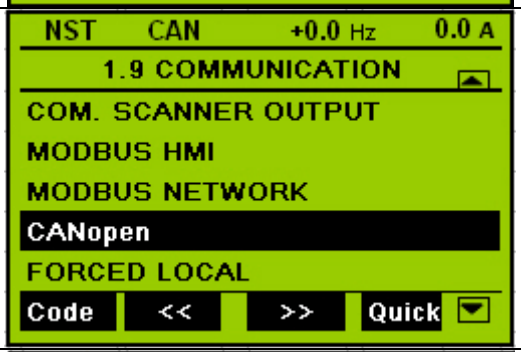
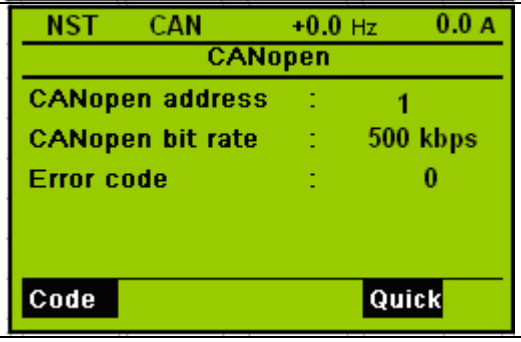
Introduction

The ATV71 parameters can be entered or modified using the graphic keypad panel.

Note

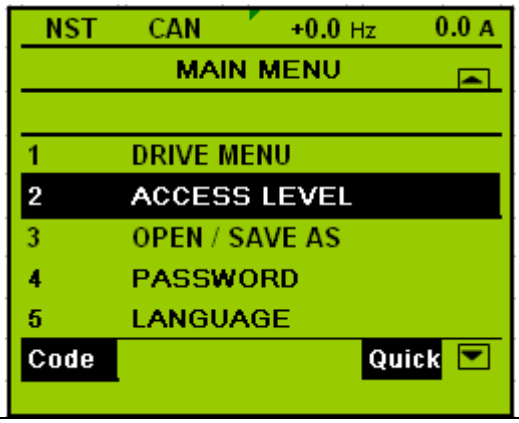
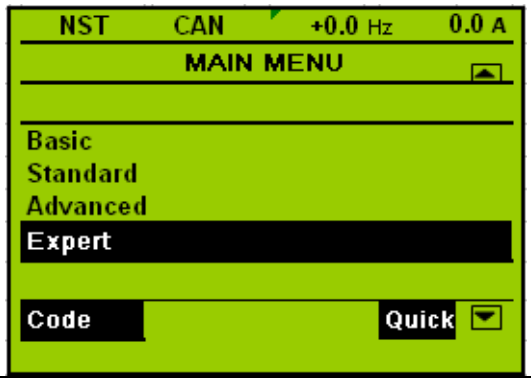
If this is not a new drive it is recommended to return to the factory settings. If you need instructions on how to do this, please read the drive documentation.

CANopen settings

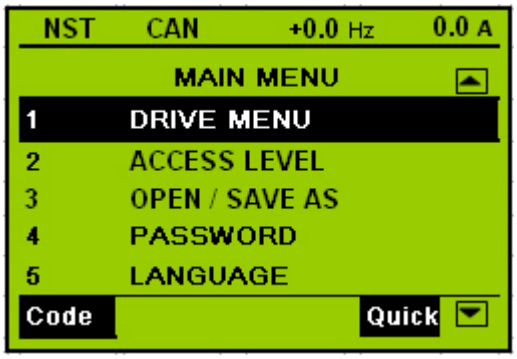
1	The CANopen address and Baudrate can be input using the jog dial on the front panel of the Altivar.	
2	To set the **CANopen address** and the **Baudrate** go to **1 DRIVE MENU** and press Enter.	
3	Go to **1.9 COMMUNICATION** and press Enter.	
4	Go to **CANopen** and press Enter.	
5	Set the **CANopen address** to **1** for the first one, and **2** for the next second one Set the **CANopen bit rate** to **500 kbps**.	

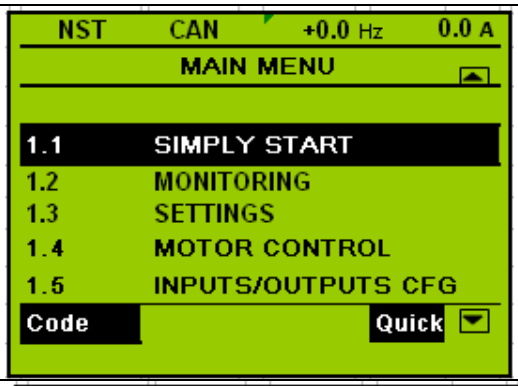
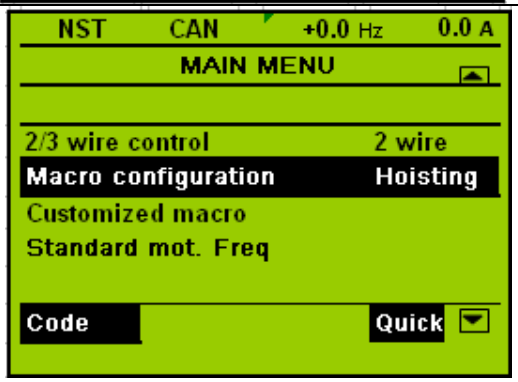
6	After changing the configuration it is necessary to power cycle the drive. **Note:** For high power drives (more than 90 kW) it is recommended to do an automatic reboot with the graphic keypad panel (refer to drive user's manual for details)	

Access level settings

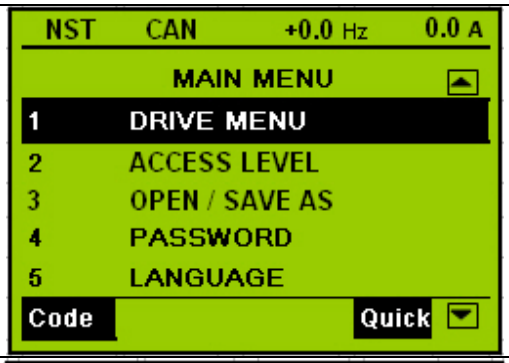
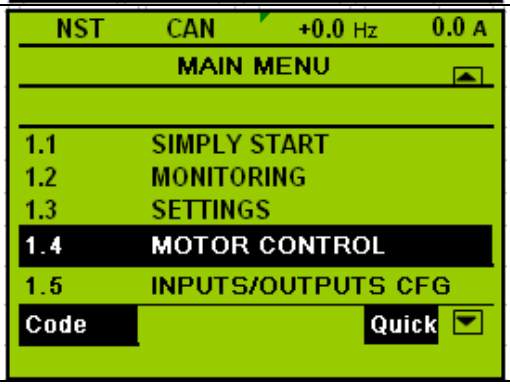
1	To change the Access level go to: **2 ACCESS LEVEL** and press Enter	
2	Go to **Expert** and press Enter	

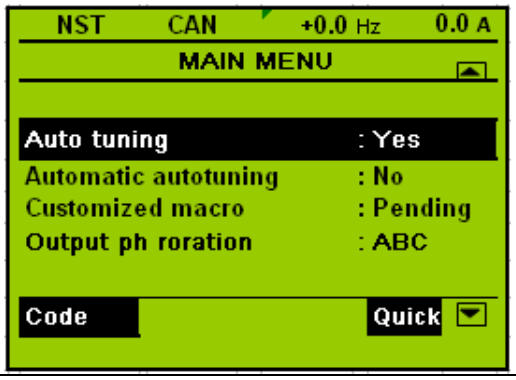
Macro configuration settings

1	To change the brake settings go to: **1 DRIVE MENU** and press Enter.	

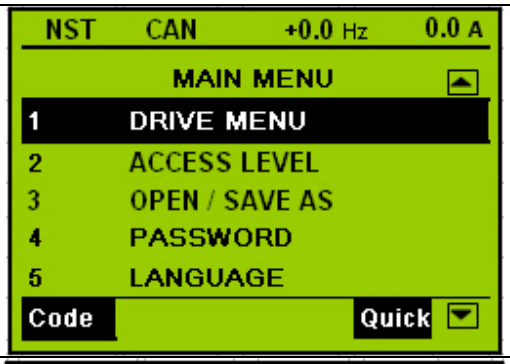
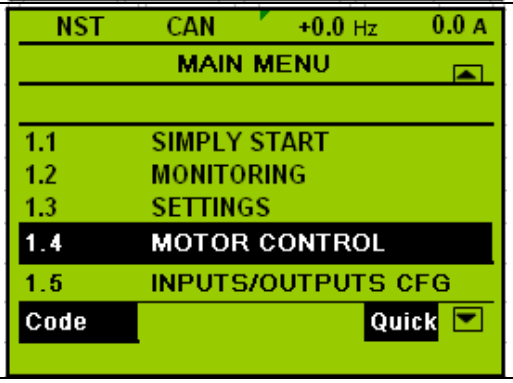
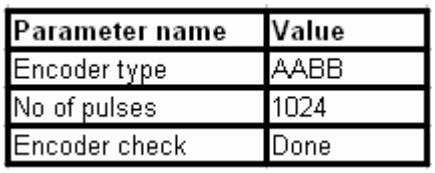
2	Go to **1.1 SIMPLY START.** and press Enter	
3	Go to **Macro configuration** and select **Hoisting** **NOTE** These parameters are for the test machine only. They are **NOT VALID** for every machine.	

Enable Auto tuning

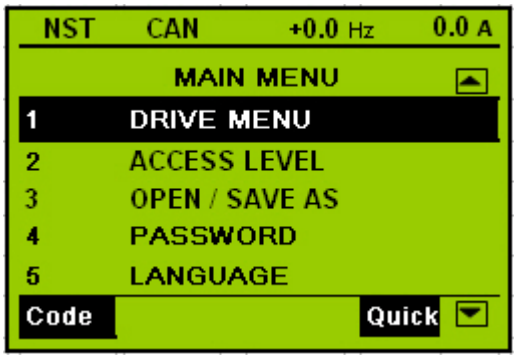
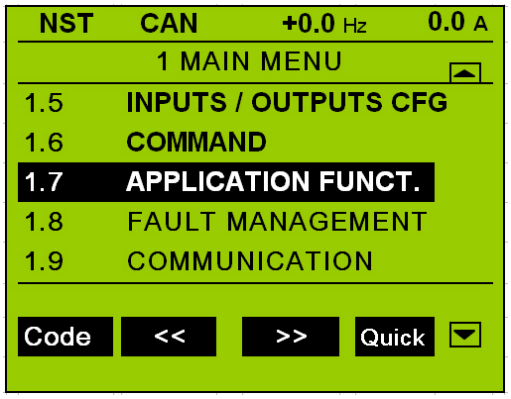
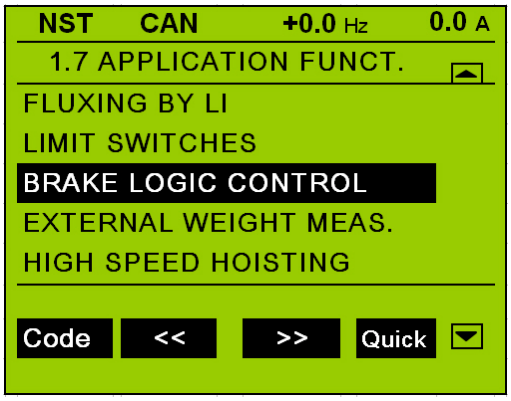
1	To enanle the auto tuning go to: **1 DRIVE MENU** and press Enter.	
2	Go to **1.4 MOTOR CONTROL** and press Enter.	

3	Go to **Auto tuning** and select **Yes** **NOTE** These parameters are for the test machine only. They are **NOT VALID** for every machine.	

Parameterization of Encoder

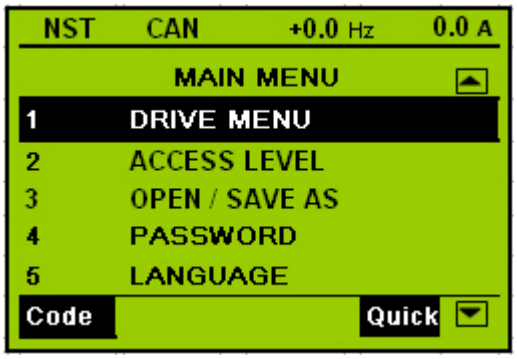
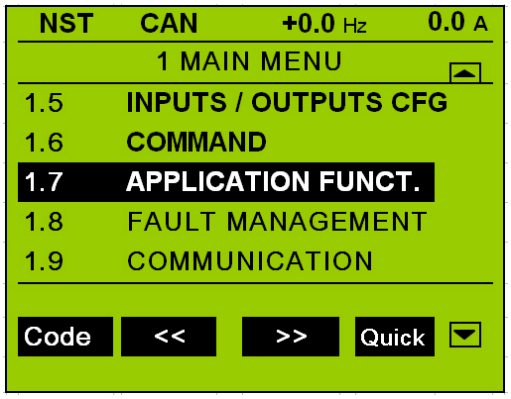
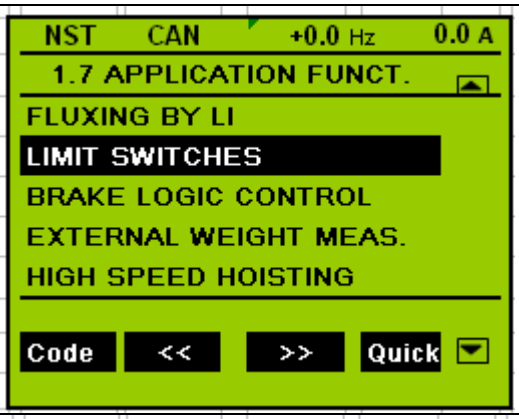
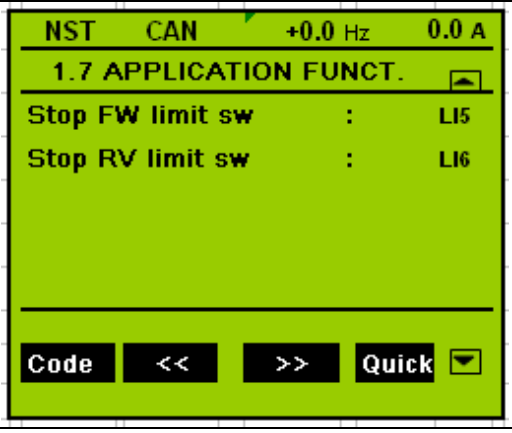
1	To parameterize the encoder go to: **1 DRIVE MENU** and press Enter.	
2	Go to **1.4 MOTOR CONTROL** and press Enter.	
3	Set the parameters to the values shown here on the right. **NOTE** These parameters are for the test machine only. They are **NOT VALID** for every machine.	

Brake settings

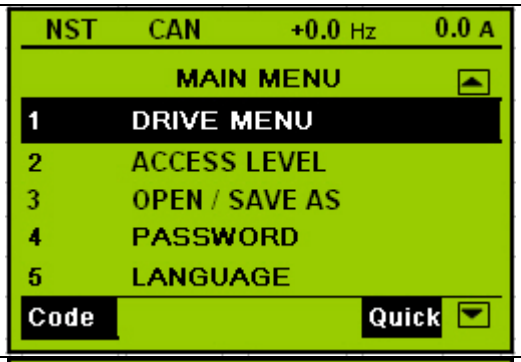
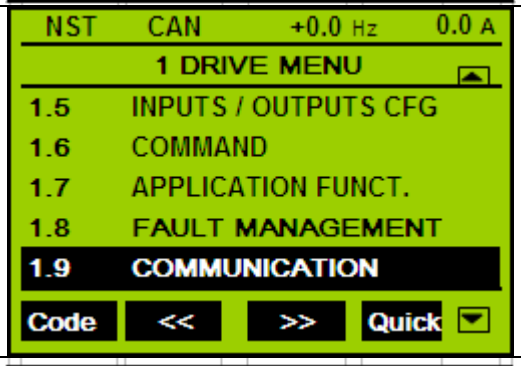
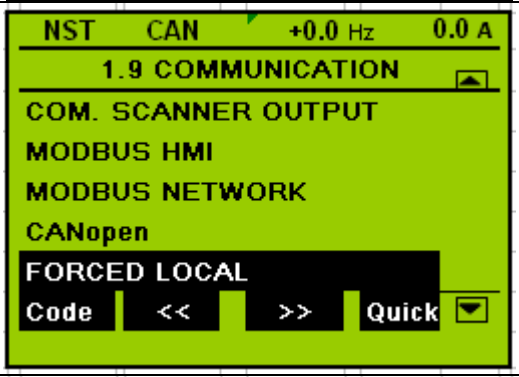
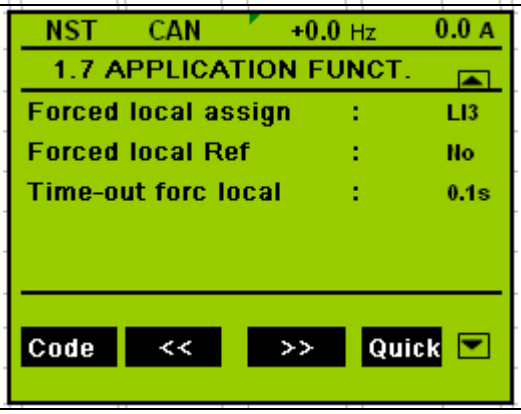
1	To change the brake settings go to: **1 DRIVE MENU** and press Enter.	
2	Go to **1.7 APPLICATION FUNCT.** and press Enter	
3	Go to **BRAKE LOGIC CONTROL** and press Enter	
4	Set the parameters to the values shown here on the right. **Note:** These parameters are for the test machine only. They are **NOT VALID** for every machine.	

Parameter name	Value
Brake assignment	R2
Movement type	Hoisting
Brake contact	No
Brake impulse	Yes
Brake release I FW	0.6
Brake release time	0.20 s
Brake release freq.	0 Hz
Brake engage time	0.20 s

Limit switch configuration

1	Limit switch configuration go to: **1 DRIVE MENU** and press Enter	
2	Go to **1.7 APPLICATION FUNCT.** and press Enter	
3	Go to **LIMIT SWITCHES** and press Enter	
4	Configure the Forward and the Reverse limit switches. **Stop FW limit sw : LI5** **Stop RV limit sw : LI6**	

Local mode configuration

1	**Local mode configuration** Go to : **1 DRIVE MENU** and press Enter	
2	Go to **1.9 COMMUNICATION** and press Enter	
3	Go to **FORCED LOCAL** and press Enter	
4	Set **Forced local assign** : **LI3** **Time-out forc local** : **0.1s**	
	Local mode: Local mode is required to manage the axis movement when the wiring is directly connected to the Altivar and not to the controller. Local mode is used to test the axis during: • Commissioning of the crane for the first time; in this case the operator is able to check that the movement is correct. • Maintenance, when it is required to move an axis without the help of the controller.	

Appendix

Detailed Component List

The following is a list of components for the main components of the Hoisting Optimized CANopen M238 architecture. A complete component list for the architecture can be found in the EPLAN file "Hoisting_Optimized_CANopen_M238_WID"

Sarel cabinet

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
1.0	1	Cabinet with mounting plate 1800 x 1000 x 600 mm (HxWxD)	NSYSF181060P	
1.1	2	Side wall 1800 x 600 mm	NSY2SP186	
1.2	1	Fan with filter, 250 m³, 230 Vac, IP54	NSYCVF165M230PF	
1.3	1	Air filter	NSYCAG125LPF	
1.4	1	Wiring diagram pocket	NSYSDP8M	
1.5	1	Cabinet light incl. socket, magnetic fixing	NSYLAM75	
1.6	1	Thermostat 1 NC, 0-60 C°	NSYCCOTHO	

Main switch

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
2.0	1	Mains witch 3pin 36 kA	LV429003	
2.1	1	Contact block TM16D	LV429035	
2.2	1	Terminal cover	LV420321	
2.3	1	Rotary drive with door interface	LV429340	

Power supply

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
3.0	1	Power supply 230 Vac / 24 Vdc, 10 A	ABL8RPS24100	
3.1	1	Circuit Breaker C60N 1P, C, 2 A	23726	
3.2	1	Circuit Breaker C60N 2P, C, 2 A	23747	
3.3	1	Circuit Breaker C60N 2P, C, 10 A	23756	
3.4	5	Circuit Breaker C60N 1P, C, 3 A	24427	
3.5	7	Circuit Breaker C60N 2P, C, 3 A	24444	
3.6	6	Auxiliary contacts for C60N	26924	

Controller

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
4.0	1	Modicon M238 Logic controller (S-type)	TM238LFDC24DTS0	
4.1	1	Digital Input Module 16 IN, 24 Vdc	TM2DDI16DT	

Drives

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
5.0	4	ATV312 variable speed drive 0.75 kW	ATV312H075N4	V5.1 IE 50
5.1	2	ATV71 variable speed drive 0.75 kW	ATV71H075N4	V2.7 IE 33
5.2	3	Brake resistor ATV71H075N4	VW3A7801	
5.3	6	Magnetic circuit breaker 4 A	GV2L08	
5.4	6	Auxiliary contacts for circuit breaker 1 NO 1 NC	GVAE11	
5.5	2	Encoder interface card Altivar 71	VW3A3401	
5.6	2	Extended I/O card Altivar 71	VW3A3202	
5.7	2	Logic I/O card Altivar 71	VW3A3201	

Sensors

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
6.0	2	Inductive Proximity Sensor OsiSense	XS618B1PBL2	
6.1	2	Photo-electric Sensor OsiSense	XUX1ARCNT16	
6.2	2	Reflector for Photo-electric Sensor OsiSense	XUZC80	
6.3	2	Inductive Proximity Sensor OsiSense	XS9C111A1M12	
6.4	6	Limit Switch OsiSense	XCKMR54D1H29	
6.5	2	Overload Limit Switch OsiSense	XCKM110	
6.6	2	Screw Limit Switch TER International (Third party)	depends on the Scaling rates	
6.7	2	Load Cell Vishay (Third party)	KISD-6	
6.8	2	Web Tension Transmitter for Load Cell	PS-1010T	

Encoder

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
7.0	2	ATV71 encoder card	VW3A3401	
7.1	2	OsiSense incremental encoder 58 mm	XCC1510PS11X	
7.2	2	Encoder cable 10 pin	XCCPM23121L5	
7.3	2	Shaft coupling with spring	XCCRAR1010	
7.4	2	Fixing brackets for 58 mm encoder	XCCRE5RN	

HMI

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
8.0	1	Magelis XBTGT 5.7“ touch display	XBTGT2330	
8.1	1	Cable for connecting XBTGT and M238	XBTZ9008	

E-Stop and Door guarding

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
9.0	2	Safety module	XPSAF5130	
9.1	1	Safety extension module	XPSECP5131	
9.2	1	Door guarding safety module XPSAC	XPSAC5121	
9.3	1	E-Stop pushbutton for field	XALK178G	
9.4	1	Emergency stop push button for cabinet door	XB5AS844	
9.5	1	Auxiliary contacts for cabinet E-Stop	ZB5AZ141	
9.6	3	Illuminated push button, 1 NC, blue	XB5AW36B5	
9.7	1	Assembly housing	XALD01	
9.8	1	Door guard switch	XCSPA792	
9.9	1	Actuator for door guard switch	XCSZ13	
9.10	2	Load contactor 18 A	LC1D18BL	
9.11	2	Auxiliary contact 2 NO+2 NC	LADN22	
9.12	10	Load contactor	LC1D09BL	

Harmony

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
10.0	1	Key lock selector switch	XB5AG41	
10.1	1	Box for 3 button	XALD03	
10.2	1	Signal lamp LED white	XB5AVB1	
10.3	2	Pushbutton with LED green	XB5AW33B5	
10.4	1	Pushbutton with LED red	XB5AW34B5	
10.5	3	Pushbutton with LED yellow	XB5AW35B5	
10.6	1	Connection element	XVBC21	
10.7	1	Signal element green	XVBC2B3	
10.8	1	Signal element red	XVBC2B4	
10.9	1	Signal element blue	XVBC2B6	
10.10	1	Signal element white	XVBC2B7	
10.11	1	Signal element white	XVBC2B5	
10.12	1	Tube with connection	XVBZ02	

CANopen

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
11.0	3	CANopen taps with 4x SUBD9	TSXCANTDM4	
11.1	1	CANopen cord set SUBD9 SUBD9 1m	TSXCANCADD1	
11.2	8	CANopen cord set SUBD9 RJ45 1 m	TCSCCN4F3M1T	
11.3	1	M12 female socket for CANopen 5 m	1525704	

Software Tools

Software-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
12.0	1	SoMachine incl. Vijeo-Designer	MSDCHNSFUV20	V2.0
12.1	1	SoMachine solution extension	MSDCHNSFUS0V20	V2.0
12.2	1	Programming cable	XBTZG935	

Component Protection Classes

Positioning Protection Class

Component	In Field, On Site			Cabinet		
				Front		Inside
	IP54	IP65	IP67	IP55	IP65	IP20
Main Switch						**X**
Emergency Stop switch housing XALK		**X**				
Preventa safety module XPS						**X**
Single/Double switch housing		**X**				
Control switch, 3 positions					**X**	
Indicator buttons		**X**				
Positions switch Universal		**X**				
Contactors						**X**
Phaseo Power Supply						**X**
Modicon M238 Logic controller						**X**
Altivar 312 and Altivar 71						**X**
TeSys contactor						**X**
Magelis XBTGT HMI					**X**	**X**

Environmental Characteristics

NOTE : The equipment represented in the architecture(s) of this document has been rigorously tested to meet the individually specified environmental characteristics for operation and storage, and that information is available in the product catalogs. If your application requirements are extreme or otherwise do not appear to correspond to the catalog information, your local Schneider Electric Support will be eager to assist you in determining what is appropriate for your particular application needs.

Component Features

Components

Compact NSX main switch

Compact NSX rotary switch disconnectors from 12 to 175 A are suitable for on-load making and breaking of resistive or mixed resistive and inductive circuits where frequent operation is required. They can also be used for direct switching of motors in utilization categories AC-3 and DC-3 specific to motors.

- 3-pole rotary switch disconnectors, 12 to 175 A
- Padlock able operating handle (padlocks not supplied)
- Degree of protection IP65

Power Supply Phaseo: ABL8RPS24100

- 1or 2-phase connection
- 100...120 Vac and 200...500 Vac input
- 24 Vdc output
- 10 A output
- Diagnostic relay
- Protected against overload and short circuits

Preventa safety module: XPSAC5121

Main technical characteristics:

For monitoring	Emergency Stop
Max. Category accord. EN 954-1	3
No. of safety circuits	3 N/O
No. of additional circuits	1 Solid-State
Indicators	2 LED
Power supply AC/DC	24 V
Response time on input opening	< 100 ms
AC-15 breaking capacity	C300
DC-13 breaking capacity	24 Vdc / 2 A - L/R 50ms
Minimum voltage and current	17 V / 10 mA
Dimensions (mm)	114 x 22.5 x 99
Connection	Captive screw-clamp terminals
Degree of protection	IP20 (terminals) IP40 (casing)

Safety modules XPSAC are used for monitoring Emergency Stop circuits conforming to standards EN ISO 13850 and EN 60204-1 and also meet the safety requirements for the electrical monitoring of switches in protection devices conforming to standard EN 1088 / ISO 14119. They provide protection for both the machine operator and the machine by immediately stopping the dangerous movement on receipt of a stop instruction from the operator, or on detection of a fault in the safety circuit itself.

Altivar 71 Variable Speed Drive

- - 200 Vac to 240 Vac 1-phase, 0.37 kW to 7.5 kW
  - 200 Vac to 240 Vac 3-phase, 0.37 kW to 75 kW
  - 380 Vac to 480 Vac 3-phase, 0.75 kW to 500 kW
  - 500 Vac to 690 Vac 3-phase, 2.2 kW to 630 kW
- Integrated EMC filter
- Temperature range: -10 to +50°C
- Speed range 0 to 1000 Hz
- Graphical display for control and parameterization
- Operation via Modbus, CANopen or other buses possible
- 2 analog inputs plus 1 analog output
- Digital inputs, 2 digital status outputs
- 1 shutdown output (Power removal function)
- Option cards for communication buses, Extended I/O and encoder
- Protections of drive and motor
- Compact design, side-by-side installation possible

Altivar 312 Variable Speed Drive

The Altivar 312 drive is a variable speed drive for 3-phase squirrel cage asynchronous motors. The Altivar 312 is robust, compact, easy to use and conforms to EN 50190, IEC/EN 61800-2, IEC/EN 61800-3 standards UL/CSA certification and to CE marking.

Altivar 312 drives communicate on Modbus and CANopen industrial buses. These two protocols are integrated as standard.

Altivar 312 drives are supplied with a heat sink for normal environments and ventilated enclosures. Multiple units can be mounted side by side to save space.

Drives are available for motor ratings between 0.18 kW and 15 kW, with four types of power supply:

- 200 Vac to 240 Vac 1-phase, 0.18 kW to 2.2 kW
- 200 Vac to 240 Vac 3-phase, 0.18 kW to 15 kW
- 380 Vac to 500 Vac 3-phase, 0.37 kW to 15 kW
- 525 Vac to 600 Vac 3-phase, 0.75 kW to 15 kW

Modicon M238 Logic controller: TM238LFDC24DTS0

The M238 is powered with 24 Vdc, offer:

- CANopen bus master link
- 14 x 24 Vdc inputs including 8 fast inputs, dedicated to special functions such as HSC high-speed counting
- 10 x 24 Vdc solid state outputs including 4 fast outputs, dedicated to special functions such as counting, PWM and PTO
- An RS 232/RS 485 serial link (ASCII or Modbus protocol).
- A Modbus RS 485 serial link mainly dedicated to connection of a Human/Machine interface terminal (link providing a 5 Vdc power supply for a Magelis Small Panel XBT NP00/R400/RT500)
- Expand the I/O count by adding up to 7 expansion modules. The following modules are available:
    - Discrete TM2 DDI/DDO/DMM/DRA
    - Analog TM2 AMI/ALM/ARI/AMO/AVO/AMM
    - High-speed counter TM200 HSC210DT/DF

Magelis HMI: XBTGT2330

- Sensor screen (STN-Technology) with 24 Vdc power supply
- Brightness and Contrast adjustment
- Communication via Uni-Telway and Modbus. Communication via Ethernet TCP/IP is also available in specific models
- Flat Profile
- Memory expansion for application program
- Temperature range: 0..+ 50 °C
- Certificates: UL, CSA

SoMachine OEM Machine Programming Software: MSDCHNSFUV20 and MSDCHNSFUS0V20

SoMachine is the OEM solution software for developing, configuring and commissioning the entire machine in a single software environment, including logic, motion control, HMI and related network automation functions.

SoMachine allows you to program and commission all the elements in Schneider Electric's Flexible and Scalable Control platform, the comprehensive solution-oriented offer for OEMs, which helps you achieve the most optimized control solution for each machine's requirements.

Flexible and Scalable Control platforms include:

Controllers:

HMI controllers:
- Magelis XBTGC HMI controller
- Magelis XBTGT HMI controller
- Magelis XBTGK HMI controller

Logic controllers:
- Modicon M238 Logic controller
- Modicon M258 Logic controller

Motion controller
- Modicon LMC058 Motion controller

Drive controller:
- Altivar ATV-IMC Drive controller

HMI:

HMI Magelis graphic panels:
- XBTGT
- XBTGK

SoMachine is a professional, efficient, and open software solution integrating Vijeo-Designer.
It integrates also the configuring and commissioning tool for motion control devices.

It features all IEC 61131-3 languages, integrated field bus configurators, expert diagnostics and debugging, as well as outstanding capabilities for maintenance and visualization.

SoMachine integrates tested, validated, documented and supported expert application libraries dedicated to Packaging, Hoisting and Conveying applications.

SoMachine provides you:
- One software package
- One project file
- One cable connection
- One download operation

Contact

Publisher	**Process & Machine Business** **OEM Application & Customer Satisfaction** Schneider Electric Automation GmbH Steinheimer Strasse 117 D - 63500 Seligenstadt Germany
Homepage	http://www.schneider-electric.com/sites/corporate/en/home.page
As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.	